

Esame Sistemi Operativi - Operating Systems Exam

2021/02/12

Ex 1 (2.5 points)

Italiano

Si supponga che il seguente programma venga eseguito con il valore 4 sulla riga di comando. Si riporti l'output generato. Si ipotizzi che il programma venga eseguito su un sistema operativo nel quale l'attesa di 1 secondo sia sufficientemente lunga per completare tutti gli altri task in esecuzione. Si prega di riportare la risposta su un'unica riga, indicando i vari messaggi e valori in output separati da un unico spazio. Non inserire nessun altro carattere nella risposta. Errori di formato verranno considerati alla stregua degli altri errori. Questo è un esempio di risposta corretta: a 4 b++ etc.

English

Suppose to execute the following program with the value 4 passed on the command line. Report the output generated by the program. Suppose to run the program in an operating system where waiting of 1 second is sufficient to allow all other tasks that are running in that specific moment of time to terminate. Please, report the response on a single line, indicating the various messages and output values separated by a single space. Do not insert any other character into the answer. Format errors will be treated in the same way as other errors. This is an example of a correct answer: a 4 b++ etc.

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>

int main(){
    int i;
    for (i=0; i<4 && !fork(); i++){
        if (fork()) {
            sleep (1);
            system ("echo i+");
        }
        execlp ("echo", "system", "i++", NULL);
    }
}
```

Risposta. Answer.

i++ i+ i++

Ex 2 (2.5 points)

Italiano

Si supponga che il seguente programma venga eseguito con il valore 4 sulla riga di comando. Si riporti l'output generato. Si prega di riportare la risposta su un'unica riga, indicando i vari messaggi e valori in output separati da un unico spazio. Non inserire nessun altro carattere nella risposta. Errori di formato verranno considerati alla stregua degli altri errori. Questo è un esempio di risposta corretta: 2 5 0 3

English

Suppose to execute the following program with the value 4 passed on the command line. Report the output generated by the program. Please, report the response on a single line, indicating the various messages and output values separated by a single space. Do not insert any other character into the answer. Format errors will be treated in the same way as other errors. This is an example of a correct answer: 2 5 0 3

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <pthread.h>
```

```

pthread_t thread;
int i;

void *t1 (void *a){
    int *p;
    p = (int *) a;
    i = *p;
    pthread_detach (pthread_self ());
    printf ("%d ", i);
    i--;
    if (i>0)
        pthread_create (&thread, NULL, t1, (void *) &i);
    return NULL;
}

int main(int argc, char **argv) {
    i = atoi (argv[1]);
    if (fork())
        pthread_create (&thread, NULL, t1, (void *) &i);
    sleep (1);
    printf ("%d ", -i);
}

```

Risposta. Answer.

4 3 2 1 0 -4

Ex 3 (2.0 points)

Italiano

Si analizzi il seguente tratto di codice.

Si indichi quali delle seguenti affermazioni sono corrette. Si osservi che risposte errate implicano una penalità nel punteggio finale.

English

Analyze the following piece of code.

Please indicate which of the following statements are correct. Note that incorrect answers imply a penalty in the final score.

```

void *thread_main(void *p) {
    int x, *y;
    y = (int *)p;
    x = *y;
    x += x;
    *p = x;
    return NULL;
}

int main() {
    int data = 1;
    pthread_t one, two;
    pthread_create(&one, NULL, thread_main, &data);
    pthread_create(&two, NULL, thread_main, &data);
    pthread_join(one, NULL);
    pthread_join(two, NULL);
    printf("%d\n", data);
    return 0;
}

```

Scegli una o più alternative. Choose one or more options.

- Il codice contiene una race-condition [The code contains a race condition](#)
- Il codice non contiene una race-condition [The code does not contain a race condition](#)
- Può essere stampato il valore 1 [The value 1 can be printed](#)
- Può essere stampato il valore 0 [The value 0 can be printed](#)
- Può essere stampato il valore 4 [The value 4 can be printed](#)
- Può essere stampato il valore 2 [The value 2 can be printed](#)

Ex 4 (1.5 points)

Italiano

Si supponga un processo diventi "orfano".

Si indichi quali delle seguenti affermazioni sono corrette. Si osservi che risposte errate implicano una penalità nel punteggio finale.

English

Suppose a process becomes an "orphan."

Please indicate which of the following statements are correct. Note that incorrect answers imply a penalty in the final score.

Scegli una o più alternative. Choose one or more options.

- Il processo sta attendendo che il padre effettui una wait [The process is waiting that the parent performs a wait](#)
- Il processo è diventato orfano in quanto non ha effettuato una wait [The process becomes an orphan because it did not perform a wait](#)
- Il processo diventerà "zombie" alla sua terminazione [The process will become "zombie" at its termination](#)
- Il processo viene ereditato dal processo "init" [The process is inherited by the "init" process](#)
- Il processo non diventerà "zombie" alla sua terminazione perché il processo "init" lo eredita [The process will not become "zombie" at its termination because the "init" process will inherit it](#)

Ex 5 (1.5 points)

Italiano

Nell'utilizzo di una pipe di nome fd, lo scrittore effettua la seguente operazione

```
write (fd[1], "Hi Child!", 9);
```

mentre il lettore effettua la lettura con il ciclo successivo:

```
char c;  
int i=0;  
while ((n = read (fd[0], &c, 1)) > 0) {  
    s[i] = c;  
    i++;  
}  
s[i] = '\n';  
print ("%s", s);
```

Si indichi quali delle seguenti affermazioni sono corrette. Si osservi che risposte errate implicano una penalità nel punteggio finale.

English

Using a pipe of name fd, the writer does the following operation:

```
write (fd[1], "Hi Child!", 9);
```

while the reader reads with the next cycle:

```
char c;  
int i=0;  
while ( (n = read (fd[0], &c, 1)) > 0) {  
    s[i] = c;  
    i++;  
}
```

```
s[i] = '\n';
print ("%s", s);
```

Please indicate which of the following statements are correct. Note that incorrect answers imply a penalty in the final score.

Scegli una o più alternative. Choose one or more options.

- Affinché l'operazione sia corretta e la stampa venga effettuata, lo scrittore deve inserire nella pipe il carattere '\n' **To have a correct execution and printing, the writer must insert the character '\n' in the pipe**
- Dato che la lettura avviene a caratteri anche la scrittura deve avvenire a caratteri **Since read is performed character-by-character, the write operation must also be performed character-by-character**
- Il lettore stampa correttamente la stringa s = "Hi child!" **The reader correctly prints the string s = "Hi child!"**
- Il lettore rimane bloccato sulla read a tempo indefinito una volta letto tutto il messaggio **The reader remains blocked on the read indefinitely after the entire message has been read**
- Affinché la stampa avvenga correttamente la lettura deve avvenire con formato while (read (fd[0], str, 9)>0); dove la variabile str viene definita come char str[10]; **To have a correct printing, the read operation must be performed with the format while (read (fd[0], str, 9)>0); where the variable str is defined as char str [10];**
- Il lettore non stampa nulla **The reader prints nothing**

Ex 6 (1.5 points)

Italiano

Si analizzi il seguente tratto di codice.

Si indichi quali delle seguenti affermazioni sono corrette. Si osservi che risposte errate implicano una penalità nel punteggio finale

English

Analyze the following piece of code.

Please indicate which of the following statements are correct. Note that incorrect answers imply a penalty in the final score.

```
if (fork() == 0) {
    /* first child */
    if (fork() == 0) {
        /* second child */
        ...
    } else {
        exit (1);
    }
}
wait ();
```

Scegli una o più alternative. Choose one or more options.

- Il padre può soffrire di deadlock **The parent can suffer of deadlock**
- Nel caso in cui il primo figlio abbia eseguito la exit(1), il secondo figlio non diventerà mai zombie **In case the first child has performed exit(1) , the second child will never become a zombie**
- Affinchè il codice sia corretto occorre inserire una seconda wait al termine **To have a correct piece of code we must insert a second wait statement at the end**
- Quando il primo figlio termina, il secondo figlio viene ereditato da "init" **When the first child terminates, the second child is inherited by "init"**
- Il padre attende la terminazione del primo figlio **The parent waits the termination of the first child**
- Il padre attende la terminazione del primo e del secondo figlio **The parent waits the termination of the first and the second child**

Ex 7 (1.5 points)

Italiano

Si faccia riferimento al termine “priority inversion”.

Si indichi quali delle seguenti affermazioni sono corrette. Si osservi che risposte errate implicano una penalità nel punteggio finale.

English

Refer to the term “priority inversion”.

Please indicate which of the following statements are correct. Note that incorrect answers imply a penalty in the final score.

Scegli una o più alternative. Choose one or more options.

- Il processo coinvolto dalla “priority inversion” è in uno stato di deadlock *The process involved in the “priority inversion” is in a deadlock state*
- Agisce a discapito di un processo a bassa priorità *It acts at the expense of a low priority process*
- Una possibile soluzione al problema è l'utilizzo del protocollo denominato “priority inheritance” *A possible solution to the problem is the use of the protocol named “priority inheritance”*
- I processi coinvolti sono almeno 2, uno ad alta priorità e uno a bassa priorità *Involved processes are at least 2, one with high priority and one with low priority*
- I processi coinvolti sono almeno 3, uno ad alta priorità, uno a media priorità e uno a bassa priorità *Involved processes are at least 3, one with high priority, one with medium priority, and one with low priority*
- Agisce a discapito di un processo ad altra priorità *It acts at the expense of a high priority process*

Ex 8 (1.5 points)

Italiano

Si faccia riferimento all'utilizzo dei segnali in ambiente UNIX/Linux.

Si indichi quali delle seguenti affermazioni sono corrette. Si osservi che risposte errate implicano una penalità nel punteggio finale.

English

Refer to the use of signals in a UNIX/Linux environment.

Indicate which of the following statements are correct. Note that incorrect answers imply a penalty in the final score.

Scegli una o più alternative. Choose one or more options.

- Attraverso la system call signal() si può decidere di ignorare QUALSIASI tipo di segnale *By using the system call signal() you can decide to ignore ANY type of signal*
- All'interno di un signal handler occorre usare solo funzioni rientranti *Inside a signal handler, only reentrant functions should be used*
- La ricezione di un segnale da parte di un processo provoca SEMPRE la terminazione del processo *The reception of a signal by a process ALWAYS causes the termination of the process*
- L'utilizzo della coppia di funzioni kill() e pause() o delle primitive semaforiche sem_post() e sem_wait() per la sincronizzazione dei processi sono EQUIVALENTI *The use of the pair of functions kill() and pause() or of the semaphore primitives sem_post() and sem_wait() are EQUIVALENT from the point of view of process synchronization*
- La ricezione di alcuni tipi di segnali (ad esempio SIGKILL) non può essere ignorata *The reception of some types of signal (for instance SIGKILL) cannot be ignored*
- L'esecuzione di un signal handler a seguito della ricezione di un segnale da parte di un processo può portare a race conditions *The execution of a signal handler after the reception of a signal by a process can lead to race conditions*

Ex 9 (1.5 points)

Italiano

Sia dato il seguente comando bash:

```
egrep -e "^202101(2[7-8]|3[01])" input.txt | egrep -e "[1-9][0-9]*\.[0-9]" | egrep -e "13\.[127]" | cut -d " " -f 2-4
```

Riportare quali righe vengono riportate in output quando il comando viene eseguito sul file input.txt riportato alla fine della domanda.

Si osservi che risposte errate implicano una penalità nel punteggio finale.

English

Given the following bash command:

```
egrep -e "^202101(2[7-8]|3[01])" input.txt | egrep -e "[1-9][0-9]*\.[0-9]" | egrep -e "13\.[127]" | cut -d " " -f 2-4
```

Report which lines are displayed when the previous command is executed on the file input.txt reported at the end of the question.

Note that incorrect answers imply a penalty in the final score.

Contenuto del file input.txt [Content of the file input.txt](#)

```
20210127 12.5 Stefano AXY
20210128 13.1 Giulia AXY
20210129 13.2 Gabriele AXY
20210130 13.3 Manuele AXY
20210131 13.7 Sara AXY
20210201 14.3 Enrico AXY
20210202 15.5 Pietro AXY
```

Scegli una o più alternative. Choose one or more options.

1. 13.3 AXY
2. 13.7 Sara AXY
3. 13.1 AXY
4. 13.7 AXY
5. 13.3 Manuele AXY
6. 13.2 AXY
7. 13.1 Giulia AXY
8. 13.2 Gabriele AXY

Ex 10 (5.0 points)

Italiano

Si scriva uno script di shell denominato x.sh che ricerchi nel sotto-albero della directory home dell'utente tutti i file con estensione .c, e rinomini con estensione .c_new solo i file che includono un stringa uguale al nome del file.

Per esempio, se solo due file sono trovati, ovvero first.c e second.c, e il file first.c include le seguenti due linee:

```
// this is first.c
main{ print "Hello\n"}
```

mentre il file second.c include queste due linee:

```
// this is my program n. 1
main{ print "Hello\n"}
```

solo il file first.c sarà rinominato e assumerà il nome first.c_new.

Si osservi che questa è una domanda di tipo aperto, con correzione manuale.

English

Write a shell script x.sh that searches in the subtree of the home directory of the user each file having extension .c, and it renames these files with extension .c_new only if the file includes a string equal to the name of the file. For example, if only two files are found, namely, first.c and second.c, and file first.c includes these two lines:

```
// this is first.c
main{ print "Hello\n"}
```

while file second.c includes these two lines:

```
// this is my program n. 1
main{ print "Hello\n"}
```

only file first.c will be renamed as first.c_new.
Note that this is an open-type question with manual correction.

Soluzione. Solution.

```
#!/bin/bash
find ~ -type f -name "*.c" > tmp.txt
while read file; do
    name=$(basename "$file")
    res=$(grep -e "$name" "$file")
    if [ ! -z "$res" ]; then
        mv "$file" "$file"_new"
    fi
done < tmp.txt
rm tmp.txt
```

Ex 11 (1.5 points)

Italiano

Si supponga di implementare il problema dei Readers e Writers utilizzando le due primitive indicate di seguito: In cui reading e writing sono variabili globali inizializzate a 0. Si indichi quali delle seguenti affermazioni sono corrette. Si osservi che risposte errate implicano una penalità nel punteggio finale.

English

Suppose to implement the problem of Readers & Writers using the two primitives that follow: In which reading and writing are global variables initialized to 0. Indicate which of the following statements are correct. Note that incorrect answers imply a penalty in the final score.

```
void read() {
    while (writing) { /*spin*/ }
    reading = 1
    // do read stuff
    reading = 0
}
void write() {
    while (reading || writing) { /*spin*/ }
    writing = 1
    // do write stuff
    writing = 0
}
```

Scegli una o più alternative. Choose one or more options.

- Readers e Writers possono accedere insieme alla sezione critica [Readers and Writers can both enter in the critical section](#)
- Readers e Writers possono soffrire di deadlock [Readers and Writers can suffer of deadlock](#)
- I Writers non accedono mai alla sezione critica [Writers never access the critical section](#)
- Più Readers possono accedere insieme alla sezione critica [Multiple Readers can enter together in the critical section](#)
- Più Writers possono accedere insieme alla sezione critica [Multiple Writers can enter together in the critical section](#)
- I Readers non accedono mai alla sezione critica [Readers never access the critical section](#)

Ex 12 (2.5 points)

Italiano

Si consideri il seguente insieme di processi:

Processo	TempoArrivo	BurstTime	Priorità
P1	0	26	2
P2	0	23	1
P3	5	15	3
P4	11	17	4
P5	13	19	5

Rappresentare mediante diagramma di Gantt l'esecuzione di tali processi utilizzando l'algoritmo Priority Scheduling (PS). Valori bassi di priorità corrispondono a priorità maggiore.

Calcolare il tempo di terminazione di ciascun processo e il tempo di attesa medio.

Si prega di riportare la risposta su un'unica riga, indicando i tempi di terminazione di P1, P2, P3, P4 e P5 seguiti dal tempo di attesa medio. Separare i numeri con un unico spazio. Riportare il tempo di attesa medio con 1 sola cifra decimale. Non inserire nessun altro carattere nella risposta. Errori di formato verranno considerati alla stregua degli altri errori. Questo è un esempio di risposta corretta: 32 101 65 76 52 47.9

English

Consider the following set of processes:

Process	ArrivalTime	BurstTime	Priority
P1	0	26	2
P2	0	23	1
P3	5	15	3
P4	11	17	4
P5	13	19	5

Represent the execution of these processes using the Priority Scheduling (PS) algorithm with the Gantt chart. Lower values of Priority correspond to higher priority.

Calculate the termination time of each job and the average wait time.

Please report the answer on a single line, indicating the termination times of P1, P2, P3, P4 and P5 followed by the average waiting time. Separate the numbers with a single space. Report the average wait time with only 1 decimal place. Do not insert any other character into the answer. Format errors will be treated in the same way as other errors. This is an example of a correct answer: 32 101 65 76 52 47.9

Fasi operative. Workflow.

PS:

```
P2: from= 0 to = 22 end = 23 waitingTime = 0 turnaroundTime = 23
P1: from= 23 to = 48 end = 49 waitingTime = 23 turnaroundTime = 49
P3: from= 49 to = 63 end = 64 waitingTime = 44 turnaroundTime = 59
P4: from= 64 to = 80 end = 81 waitingTime = 53 turnaroundTime = 70
P5: from= 81 to = 99 end = 100 waitingTime = 68 turnaroundTime = 87
```

EndTime: 23 49 64 81 100

AverageWaitingTime: (0 + 23 + 44 + 53 + 68) / 5 = 37.60

AverageTournaroudTime: (23 + 49 + 59 + 70 + 87) / 5 = 115.20

GanttChart:

```
2222222222222222222222221111111111111111111111111111111111111111111133333333333333334444444444444444555
555555555555555555
```

Risposta. Answer

23 49 64 81 100 37.6

Ex 13 (2.5 points)

Italiano

Si faccia riferimento all scheduling dei processi.

Si indichi quali delle seguenti affermazioni sono corrette. Si osservi che risposte errate implicano una penalità nel punteggio finale.

English

Refer to the scheduling of processes.

Indicate which of the following statements are correct. Note that incorrect answers imply a penalty in the final score.

Scegli una o più alternative. Choose one or more options.

- La strategia SRTF usa la prelazione per la schedulazione dei processi [The strategy SRTF uses preemption for processes scheduling](#)
- Nel caso di assenza di prelazione, i processi verranno eseguiti fino a quando non vorranno utilizzare ulteriormente la CPU [Without preemption, processes will run until they do not want to use the CPU any further](#)
- Un processo è inserito nella coda ready quando è in grado di utilizzare la CPU [A process is placed in the ready queue when it is able to use a CPU](#)
- Il "response time" è il tempo che trascorre da quando il processo arriva (è sottomesso) a quando esso finisce [The "response time" is the time from when the process arrives \(it is submitted\) to when it ends](#)
- "turnaround time" è il tempo che trascorre da quando il processo arriva (è sottomesso) a quando la CPU inizia a eseguirlo [The "turnaround time" is the time that it takes from when the process arrives \(it is submitted\) to when the CPU starts working on it](#)
- La strategia SJF usa la prelazione per la schedulazione dei processi [The strategy SJF uses preemption for processes scheduling](#)

Ex 14 (5.0 points)

Italiano

Un programma è costituito da più thread, e da una variabile di stato condivisa che indica uno tra due possibili stati. Tali stati sono denominati RED e GREEN. La funzione `change()` eseguita da un thread permette di modificare la variabile di stato, quindi la funzione `change()` chiamata quando lo stato è GREEN imposta la variabile di stato condivisa allo stato RED e viceversa. La funzione `red()` non è bloccata quando lo stato è RED, è invece bloccante quando lo stato è GREEN. Similmente, la funzione `green()` non è bloccante quando lo stato è GREEN ma è bloccante quando lo stato è RED. Si osservi che le funzioni `change()`, `red()` e `green()` possono essere eseguite da un qualsiasi numero di thread.

Realizzare le funzioni `change()`, `red()` e `green()` con il numero minimo di semafori POSIX, definendone le relative variabili condivise. Si ipotizzi che lo stato iniziale sia RED per tutti i thread.

Esempio di esecuzione:

T1 chiama `red()` e non si blocca.

T2 chiama `red()` e non si blocca.

T3 chiama `green()` e si blocca.

T4 chiama `green()` e si blocca.

T1 chiama `red()` e non si blocca.

T5 chiama `change()`, lo stato cambia in GREEN e T3 e T4 sono sbloccati.

T5 chiama `green()` e non si blocca.

Si osservi che questa è una domanda di tipo aperto, con correzione manuale.

English

A program includes more threads, and a shared state variable that indicates one of two possible states. These states are called RED and GREEN. The function `change()` allows a thread to modify the state variable; thus if a thread calls `change()` when the state is GREEN it changes the shared state variable to RED, and vice-versa. The `red()` function does not block a thread when the state is RED, but it blocks a thread when the state is GREEN. Similarly, the `green()` function does not block a thread when the state is GREEN, but it blocks a thread when the state is RED. Notice that function `change()`, `red()`, and `green()` can be executed by any number of threads.

Implement functions `change()`, `red()`, and `green()` with the minimum number of POSIX semaphores, defining all needed shared variables. Assume that the initial state is RED for all threads.

Note that this is an open question, with manual correction.

Example of execution:

T1 calls `red()` and it does not block.

T2 calls `red()` and it does not block.

T3 calls green() and it blocks.

T4 calls green() and it blocks.

T1 calls red() and it does not block.

T5 calls change(), the status changes to GREEN, and T3 and T4 are released (unblocked).

T5 calls green() and it does not block.

Soluzione. Solution.

```
#define RED 0
#define GREEN 1

int state = RED;

sem_t s_red;
sem_t s_green;
sem_t m;

sem_init(&s_red, 0, 1);
sem_init(&s_green, 0, 0);
sem_init(&m, 0, 1);

red(){
    sem_wait(&s_red);
    sem_post(&s_red);
}
green(){
    sem_wait(&s_green);
    sem_post(&s_green);
}
change(){
    sem_wait(&m);
    if(state == RED){
        state = GREEN;
        sem_wait(&s_red);
        sem_post(&s_green);
    }else{
        state = PING;
        sem_wait(&s_green);
        sem_post(&s_red);
    }
    sem_post(&m);
}
```

Ex 15 (1.5 points)

Italiano

Si consideri il fenomeno dello stallo o deadlock.

Si indichi quali delle seguenti affermazioni sono corrette. Si osservi che risposte errate implicano una penalità nel punteggio finale.

English

Refer to deadlock.

Indicate which of the following statements are correct. Note that incorrect answers imply a penalty in the final score.

Scegli una o più alternative. Choose one or more options.

- Se tutte le condizioni necessarie relative al deadlock sono verificate allora SICURAMENTE il sistema è in deadlock *If all necessary conditions referred to deadlock are verified, then CERTAINLY the system is in a deadlock state*
- Se solo una condizione necessaria relativa al deadlock non è verificata allora SICURAMENTE il sistema NON è in deadlock *If only one necessary condition referred to deadlock is not verified, then CERTAINLY the system is NOT in a deadlock state*
- La "starvation" è una condizione necessaria *The "starvation" is a necessary condition*
- La "attesa circolare" NON è una condizione necessaria *The "circular wait" is NOT a necessary condition*
- La "hold and wait" (possessione e attesa) è una condizione necessaria *The "hold and wait" is a necessary condition*
- Se tutte le condizioni necessarie relative al deadlock sono verificate allora il sistema PUO' essere in deadlock *If all necessary conditions referred to deadlock are verified, then the system COULD be in a deadlock state*

Ex 16 (3.0 points)

Italiano

Sia dato un sistema con 4 processi (P1, P2, P3, P4) e 3 tipi di risorse (R1, R2, R3), in cui vi siano 2 istanze di risorse di tipo R1 e 1 singola istanza per le risorse di tipo R2 e R3. Lo stato del sistema è il seguente:

- Il processo P1 detiene una risorsa {R1}
- Il processo P2 detiene una risorse {R1} ed è in attesa delle risorse {R2, R3}
- Il processo P3 detiene la risorsa {R2} ed è in attesa di una risorsa {R1}
- Il processo P4 detiene la risorsa {R3} ed è in attesa della risorsa {R2}

Si effettui un'analisi della situazione dal punto di vista del deadlock basandosi ESCLUSIVAMENTE sulla presenza di cicli nel grafo di allocazione delle risorse.

Si indichi quali delle seguenti affermazioni sono corrette. Si osservi che risposte errate implicano una penalità nel punteggio finale.

English

A system has 4 processes (P1, P2, P3, P4) and 3 types of resources (R1, R2, R3), in which the number of instances of resources of type R1 is 2, and the number of instances for each resources of type R2 and R3 is 1. The state of the system is the following:

- Process P1 holds resource {R1}
- Process P2 holds resource {R1} and it waits resources {R2, R3}
- Process P3 holds resource {R2} and it waits resource {R1}
- Process P4 holds resource {R3} and it waits resource {R2}

Analyze the situation from the point of view of the deadlock, focusing the analysis ONLY on the presence of cycles in the resource allocation graph.

Please indicate which of the following statements are correct. Note that incorrect answers imply a penalty in the final score.

Scegli una o più alternative. Choose one or more options.

- Il sistema SICURAMENTE NON si è in una condizione di deadlock *The system SURELY is NOT in a deadlock condition*
- Eliminando l'arco R2->P3 si ha la CERTEZZA di NON essere in una condizione di deadlock *Eliminating the edge R2->P3, SURELY the system is NOT in a deadlock condition*
- Eliminando l'arco R3->P4 si ha la CERTEZZA di NON essere in una condizione di deadlock *Eliminating the edge R3->P4, SURELY the system is NOT in a deadlock condition*
- Il deadlock è POSSIBILE *The deadlock is POSSIBLE*
- Eliminando l'arco P2->R2 si ha la CERTEZZA di NON essere in una condizione di deadlock *Eliminating the edge P2->R2, SURELY the system is NOT in a deadlock condition*
- Il sistema SICURAMENTE è in una condizione di deadlock *The system is SURELY in a deadlock condition*