

INFORMATICA III – prova del 14 gennaio 2008 - A

NOME _____

1 Macchina di Turing

Scrivere le istruzioni (stato, simbolo letto, simbolo scritto, movimento testina, nuovo stato) di una macchina di Turing deterministica con alfabeto (s0, 0, 1) che, preso in input un numero binario sulnastro lo incrementa di 2. Assumi pure che il numero sia a sinistra rispetto la posizione iniziale della testina.

2 Dangling pointers

Il seguente programma C (ignora include mancanti) è corretto? Mostra l'output atteso? Come lo modifichereesti?

```
int *example () {
    int A[5] = {1, 2, 3, 4, 5};
    return(A);
}

void main () {
    int i;
    int *B = example();

    for (i=0; i<5; i++)
        printf("%d\n", B[i]);
}
```

3 Cyclone

Come riscriveresti il codice del punto 2 con Cicolone? Quali altre alternative avresti?

4 Record di attivazione

Considera il seguente programma:

```
#include <iostream.h>

void print_facts(int num1, int num2);
int max_of_two(int j, int k);
double avg_of_two(int c, int d);

int main(){
    int i;
    int j;
    // point 1
    i = -8;
    j = 7;
    // point 2
    print_facts(i, j);
    // point 10
    return 0;
}

void print_facts(int num1, int num2){
    int larger;
```

```
double the_avg;
// point 3
larger = max_of_two(num1, num2);
// point 6
the_avg = avg_of_two(num1, num2);
// point 9
cout << "For the two integers " << num1
    << " and " << num2 << ',' << endl;
cout << "the larger is " << larger
    << " and the average is "
    << the_avg << '.' << endl;
}

int max_of_two(int j, int k){
    // point 4
    if (j < k)
        j = k;
    // point 5
    return j;
}

double avg_of_two(int c, int d){
    double sum;
```

```

// point 7
sum = c + d;
// point 8
return (c + d) / 2.0;
}

```

Disegna con esattezza i record d'attivazione (con tutti i link tranne il Return address) per il main nei diversi punti con commenti //point.

5 Passaggio parametri

Considera il seguente codice C++:

```

#include < stdio.h >
int x=0;
void p(int,int);
void main(){
    int x = 1;
    p(x,x);
}
void p(int y, int z){
    x = x+1;
    y = y+1;
    z = z+1;
    printf("%d\n",x+y+z);
}

```

1. Cosa stampa?
2. Se modifichiamo la dichiarazione di p passando i parametri per riferimento, `p(int &y, int &z) { ... }`, quale è il valore stampato?

6 OO

Ereditarietà e sottotipazione non coincidono. Quale è la differenza? In un linguaggio di prog. a tua scelta, fai (a) un esempio di sottotipo che non è una sottoclasse e (b) di una sottoclasse che non è un sottotipo (se il linguaggio non permette un esempio – oppure spiega perchè non è possibile).

7 Dynamic Binding in Java

Date le seguenti dichiarazioni:

```

class Pizza{
    public boolean equals(Pizza p){
        System.out.println("P");
        return true;
    }
}

class PizzaMargherita extends Pizza{
    public boolean equals(PizzaMargherita
1){
        System.out.println("M");
        return true;
    }
}

```

dire con precisione quali di TUTTE le seguenti istruzioni sono corrette e quale l'output da esse prodotto (se c'è un errore, scrivi **ERRORE**, ignoralo e continua, se l'istruzione è corretta ma non ha output scrivi **OK**) – spiegando bene (cioè anche il processo di early e late binding dove necessario) il perchè:

	OUTPUT	MOTIVO
Object o = new Pizza();		
Pizza p = new Pizza();		
Pizza p2 = o;		

PizzaMargherita l = new PizzaMargherita();		
Pizza p3 = new PizzaMargherita();		
p.equals(p);		
p.equals (o);		
p3.equals (p3);		
p3.equals(l);		
l.equals(p3);		
l.equals(l);		

8 C++

Completare il codice sottostante in tre modi diversi affinché si abbiano i seguenti output su schermo:

- 1) 9
 9
- 2) 10
 20
- 3) 9
 20

(n.b: il numero di linee non corrisponde necessariamente al numero di righe di codice)

```
#include <iostream>
#include <stdlib.h>

using namespace std;

class CPolygon {
protected:
    int width, height;
public:
    void set_values (int a, int b)
        { width=a; height=b; }

```

```

    void printarea (void)
    { cout << this->area() << endl; }
};

class CRectangle: public CPolygon {
public:
    int area (void)
    { return (width * height); }
};

class CTriangle: public CPolygon {
public:
    int area (void)
    { return (width * height / 2); }
};

int main () {


---




---




---



    ppoly1->set_values (4,5);
    ppoly2->set_values (4,5);
    ppoly1->printarea();
    ppoly2->printarea();
    delete ppoly1;
    delete ppoly2;

    system("PAUSE");
    return 0;
}
}

```

9 Semantica Assiomatica

Fai un sempio di applicazione della regola condizionale su un programma completo di pre e post condizioni a tua scelta.

INFORMATICA III – prova del 14 gennaio 2008 - B

NOME _____

1 Macchina di Turing

Scrivere le istruzioni (stato, simbolo letto, simbolo scritto, movimento testina, nuovo stato) di una macchina di Turing deterministica con alfabeto (s0, 0, 1) che, preso in input un numero binario sulnastro lo incrementa di 2. Assumi pure che il numero sia a sinistra rispetto la posizione iniziale della testina.

2 Dangling pointers

Il seguente programma C (ignora include mancanti) è corretto? Mostra l'output atteso? Come lo modifichereesti?

```
int *example () {
    int A[5] = {1, 2, 3, 4, 5};
    return(A);
}

void main () {
    int i;
    int *B = example();

    for (i=0; i<5; i++)
        printf("%d\n", B[i]);
}
```

3 Cyclone

Come riscriveresti il codice del punto 2 con Cicolone? Quali altre alternative avresti?

4 Record di attivazione

Considera il seguente programma:

```
#include <iostream.h>

void print_facts(int num1, int num2);
int max_of_two(int j, int k);
double avg_of_two(int c, int d);

int main(){
    int i;
    int j;
    // point 1
    i = -8;
    j = 7;
    // point 2
    print_facts(i, j);
    // point 10
    return 0;
}

void print_facts(int num1, int num2){
    int larger;
```

```
double the_avg;
// point 3
larger = max_of_two(num1, num2);
// point 6
the_avg = avg_of_two(num1, num2);
// point 9
cout << "For the two integers " << num1
    << " and " << num2 << ',' << endl;
cout << "the larger is " << larger
    << " and the average is "
    << the_avg << '.' << endl;
}

int max_of_two(int j, int k){
    // point 4
    if (j < k)
        j = k;
    // point 5
    return j;
}

double avg_of_two(int c, int d){
    double sum;
```

```

// point 7
sum = c + d;
// point 8
return (c + d) / 2.0;
}

```

Disegna con esattezza i record d'attivazione (con tutti i link tranne il Return address) per il main nei diversi punti con commenti //point.

5 Passaggio parametri

Considera il seguente codice C++:

```

#include < stdio.h >
int x=0;
void p(int,int);
void main(){
    int x = 3;
    p(x,x);
}
void p(int y, int z){
    x = x+1;
    y = y+1;
    z = z+1;
    printf("%d\n",x+y+z);
}

```

3. Cosa stampa?

4. Se modificiamo la dichiarazione di p passando i parametri per riferimento, `p(int &y, int &z) { ... }`, quale è il valore stampato?

6 OO

Ereditarietà e sottotipazione non coincidono. Quale è la differenza? In un linguaggio di prog. a tua scelta, fai (a) un esempio di sottotipo che non è una sottoclasse e (b) di una sottoclasse che non è un sottotipo (se il linguaggio non permette un esempio – oppure spiega perchè non è possibile).

7 Dynamic Binding in Java

Date le seguenti dichiarazioni:

```

class Pizza{
    public boolean equals(Pizza p){
        System.out.println("A");
        return true;
    }
}

```

```

class PizzaMargherita extends Pizza{
    public boolean equals(PizzaMargherita
1){
        System.out.println("B");
        return true;
    }
}

```

dire con precisione quali di TUTTE le seguenti istruzioni sono corrette e quale l'output da esse prodotto (se c'è un errore, scrivi **ERRORE**, ignoralo e continua, se l'istruzione è corretta ma non ha output scrivi **OK**) – spiegando bene (cioè anche il processo di early e late binding dove necessario) il perchè:

	OUTPUT	MOTIVO
Object o = new Pizza();		
Pizza p = new Pizza();		
Pizza p2 = o;		

PizzaMargherita l = new PizzaMargherita();		
Pizza p3 = new PizzaMargherita();		
p.equals(p);		
p.equals (o);		
p3.equals (p3);		
p3.equals(l);		
l.equals(p3);		
l.equals(l);		

8 C++

Completare il codice sottostante in tre modi diversi affinché si abbiano i seguenti output su schermo:

- 1) 9
 9
- 2) 10
 20
- 3) 9
 20

(n.b: il numero di linee non corrisponde necessariamente al numero di righe di codice)

```
#include <iostream>
#include <stdlib.h>

using namespace std;

class CPolygon {
protected:
    int width, height;
public:
    void set_values (int a, int b)
        { width=a; height=b; }
};

class CRectangle: public CPolygon {
public:
    int area (void)
        { return (width * height); }
};

class CTriangle: public CPolygon {
public:
    int area (void)
        { return (width * height / 2); }
};

int main () {
};
```

```
ppoly1->set_values (4,5);
ppoly2->set_values (4,5);
ppoly1->printarea();
ppoly2->printarea();
delete ppoly1;
delete ppoly2;

system("PAUSE");
return 0;
}
```

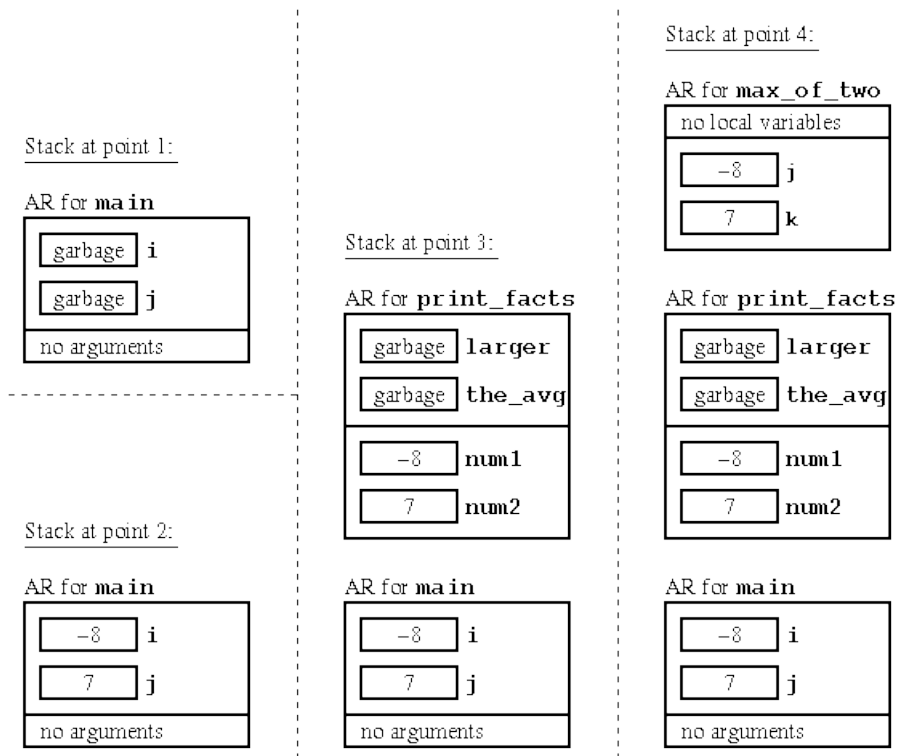
9 Semantica Assiomatica

Fai un esempio di applicazione della regola condizionale su un programma completo di pre e post condizioni a tua scelta.

soluzione

Example program: diagrams for points 1, 2, 3 and 4

[\[back to code listing\]](#)

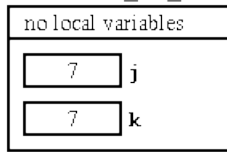


Example program: diagrams for points 5, 6 and 7

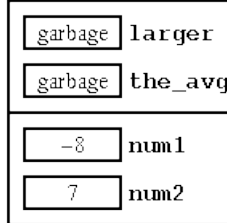
[\[back to code listing\]](#)

Stack at point 5:

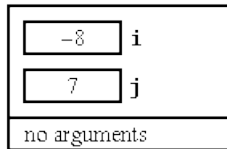
AR for `max_of_two`



AR for `print_facts`

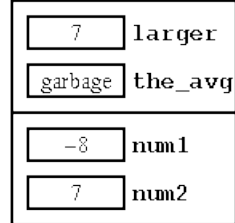


AR for `main`

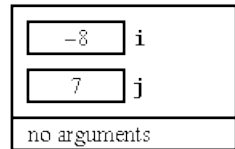


Stack at point 6:

AR for `print_facts`

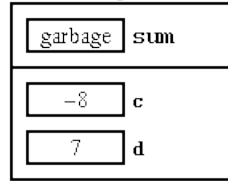


AR for `main`

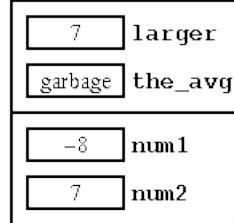


Stack at point 7:

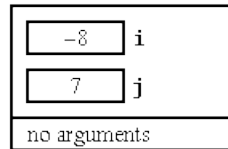
AR for `avg_of_two`



AR for `print_facts`



AR for `main`



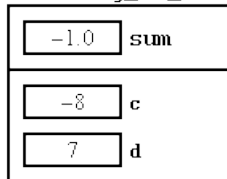
[\[back to top of document\]](#)

Example program: diagrams for points 8, 9 and 10

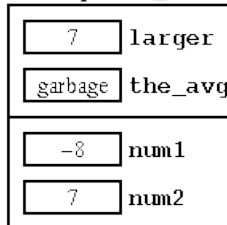
[\[back to code listing\]](#)

Stack at point 8:

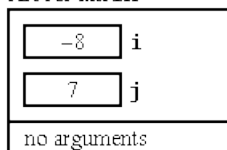
AR for `avg_of_two`



AR for `print_facts`

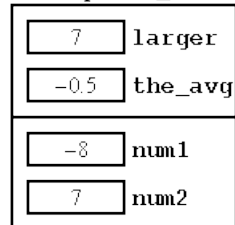


AR for `main`

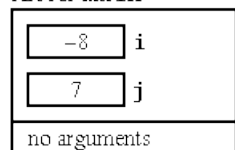


Stack at point 9:

AR for `print_facts`

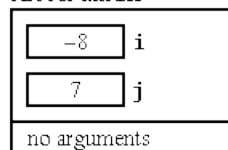


AR for `main`



Stack at point 10:

AR for `main`



passaggio paramteri

Solution

1. The value printed is 5 (Since C++ has static scope, the x in the body of p is the global variable, which has value 1 when the print instruction is executed.)
2. With call-by-reference the value printed is 7: at the moment in which the print instruction is executed, in fact, x has still value 1, while y and z point to the x local in main and have value 3.