

IJVM - Compito 2

Istruzioni: Seguire attentamente queste istruzioni per fare correttamente l'esame:

- Scaricare l'elaborato iniziale *esame.jas* dal modulo per la consegna, dovete usare questo file come punto iniziale del vostro progetto.
- Nel sorgente troverete i metodi per la moltiplicazione, per la divisione, per il modulo e per gestire l'input ed output di numeri in notazione decimale. Nello specifico potete prendere in input un numero con:

```
LDC_W OBJREF
INVOKEVIRTUAL input
```

Il numero letto in input verrà memorizzato nello stack, e quindi inserendo una `ISTORE` subito dopo l'`INVOKEVIRTUAL` potete salvare il numero letto in una variabile. Ogni input nella input console va diviso da un ";", quindi per prendere in input un numero dovete inserire "12;", per inserire due numeri "12;14;" e così via. Ricordatevi che se dovete prendere in input due numeri, allora dovete richiamare la funzione input due volte. Potete stampare un numero in output con:

```
LDC_W OBJREF
BIPUSH 0x4
INVOKEVIRTUAL print
```

Dove `BIPUSH` può essere sostituito con una `ILOAD` se volete stampare una variabile o con qualunque istruzione credete sia necessaria. Non è detto che il programma dato funzioni per tutti gli input, potreste trovare certe combinazioni di valori nelle quali il programma va in loop infinito, come per gli anni passati l'importante è scrivere correttamente il programma, non basatevi troppo sull'input e gli output. Per questo motivo vi consiglio di fare una prima soluzione senza gestire input ed output, ed in un secondo momento, quando il programma è funzionante, aggiungere i metodi per gestirli.

- Rinominate il vostro programma *NOME_COGNOME_MATRICOLOLA_IJVM.txt* e caricatelo, prima della scadenza, nella pagina su elearning. Se la vostra matricola contiene caratteri, toglieteli e lasciate solo i numeri (ES. 23ES2423 diventerà 232423).
- Il vostro programma verrà compilato ed eseguito usando una serie di input decisi da noi, se per qualche input il valore in output non è corretto, il vostro programma sarà classificato come errato. Per questo motivo è cruciale utilizzare correttamente le funzioni di input e di output.
- Se durante l'inserimento dell'input, sbagliate un numero, non cancellatelo e premete reset per reinserirlo da capo.
- Se eseguite un programma errato, proteste andare a modificare il programma caricato nella macchina virtuale, se ottenete risultati strani, riassemblete il programma e ricaricatelo.

Convertire il seguente programma in IJVM

```
1 fun1 (a, b) {
2     if(a % 2 == 0){
3         while(a-1 > 3 + b){
4             a--;
5         }
6     } else {
7         if(a >= b/2){
8             a = a+3;
9         }
10    }
11
12    return a;
13 }
14
15 fun2(b, c){
16    g = 10;
17
18    if(b > g + c){
19        while(g > 2){
20            c = b + g/2;
21            g--;
22        }
23    } else {
24        b = b/2;
25    }
26
27    return (b+c)/3;
28 }
29
30 main(){
31    int a, b, c;
32
33    input a;
34    input b;
35    input c;
36
37    print fun1(a, b) + fun1(c, fun2(a, b))
38 }
```

Esempio:

a = 5
b = 4
c = 8
output = 14