

IJVM - Compito 0

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_MATRICOLA_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 (num) {
2     steps = 0;
3
4     while (num != 0)
5     {
6         if (num % 2 == 0)
7         {
8             num /= 2;
9         }
10        else
11        {
12            num--;
13        }
14
15        steps++;
16    }
17
18    return steps;
19 }
20
21 fun2(x, y){
22     if (x + 2 > y) {
23         y = x+y+fun1(x-y);
24     } else {
25         y = x+y;
26     }
27
28     y++;
29
30     return y;
31 }
32
33 main(){
34     input x;
35     input y;
36
37     a = fun1(x);
38     b = fun2(x, y);
39
40
41     print (a+b)*2;
42 }
```

Esempio:

x = 3
y = 5
output = 24