Problem 1:
The pros and cons of having variable be initialized upon declaration are many, and depend heavily on the situation. One easily identifiable pro is that no variable will be declared and left uninitialized. This can help prevent programmers from declaring variables and forgetting what they were going to use it for. An example would be:

```
Int a = 10;
Int b,c;
for(int i = 0; i < a; i++) {
B +=i;
}
```

Another pro is that this style improves readability. By forcing variables to be instantiated when used, the first time you see the variable, it is assigned a value. This can help with reading as you know what type the variable is and how its initially being used right off the bat, especially if the variable names are not descriptive.

A major con is that this can be easily bypassed by initializing the variables to 0 and then updating them as needed. This would take up more memory than just leaving the variable uninitialized. An example would be

```
Int a = 0;
Int b = 0;
String s = “”;

for(int i = 0; i < 10; i++) {
        a+= 2;
        B ++;
s+= “hello “;
}
```

Another major con is that it makes it hard to restructure code. When you can declare a variable without initializing it, it provides the freedom to use it where you want. Having to initialize it when you use it makes this difficult, especially if you don't know where the variable is going to be used, or when it will be needed. I would rate this idea as a 4/10. Due to its major flaws that could allow its intended use to be bypassed, while not inherently bad, i would recommend against it.

Problem 2:

Pt2.

```
freedmjs pts/0        75.187.67.92     Tue Feb 11 01:48   still logged in
zmudam   pts/20       65.189.26.105    Sun Feb  9 19:38 - 19:38  (00:00)
zmudam   pts/7        65.189.26.105    Sun Feb  9 19:29 - 19:31  (00:01)
zmudam   pts/25       10.33.3.116      Fri Feb  7 13:19 - 14:37  (01:17)
zmudam   pts/15       172.25.53.109    Fri Feb  7 13:07 - 17:25  (04:18)
zmudam   pts/6        10.33.3.116      Fri Feb  7 11:33 - 13:02  (01:29)
zmudam   pts/9        65.189.26.105    Thu Feb  6 18:47 - 18:49  (00:01)
zmudam   pts/53       10.33.3.116      Wed Feb  5 13:50 - 14:41  (00:51)
zmudam   pts/52       10.33.3.116      Wed Feb  5 13:50 - 14:41  (00:51)
freedmjs pts/49       172.25.54.25     Wed Feb  5 13:49 - 13:53  (00:04)
```

Problem 3:

Everything should work.

The Z+- program 1:

```
INT i ;
i = 1 ;
FOR 1000 FOR 10000 i += 1 ; ENDFOR ENDFOR
```

The Z+- program 2:

```
INT i ;

i = 1 ;
FOR 100  i += 1 ; ENDFOR
FOR 100  i += 1 ; ENDFOR
FOR 100  i += 1 ; ENDFOR
FOR 100  i += 1 ; ENDFOR
FOR 100  i += 1 ; ENDFOR
FOR 100  i += 1 ; ENDFOR
FOR 100  i += 1 ; ENDFOR
FOR 100  i += 1 ; ENDFOR
FOR 100  i += 1 ; ENDFOR
FOR 100  i += 1 ; ENDFOR
```

The Z+- program 3:

```
String x ;
x = "hello" ;
FOR 100 x += "hello" ; ENDFOR
```

Z+-	Java
5.088293699	0.0034385
0.0675431	7.001E-6
0.0407731	2.455E-4

My findings are normal of a interpreted language because they have to convert to another language before being compiled, whereas compiled languages just compile, thereby making them run faster.