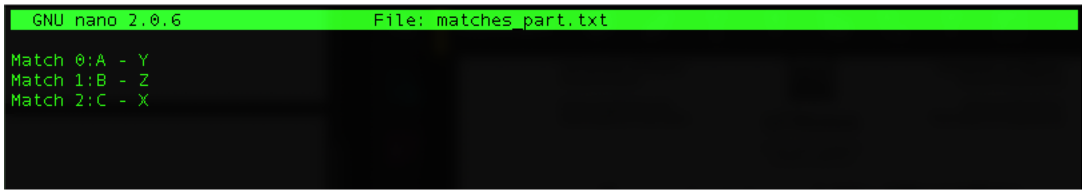
Stefan Todorovic 300118046

1.



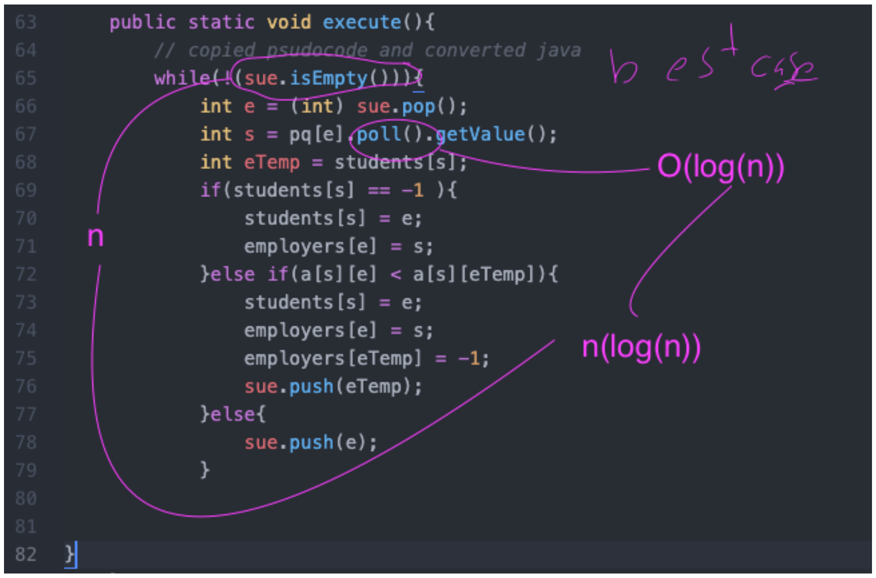
```
GNU nano 2.0.6                File: matches_part.txt

Match 0:A - Y
Match 1:B - Z
Match 2:C - X
```

Solution 1 is correct.
Each company will receive their favorite student

2.

Best run time:



```
    public static void execute(){
        // copied psudocode and converted java
        while(!(sue.isEmpty())){
            int e = (int) sue.pop();
            int s = pq[e].poll().getValue();
            int eTemp = students[s];
            if(students[s] == -1 ){
                students[s] = e;
                employers[e] = s;
            }else if(a[s][e] < a[s][eTemp]){
                students[s] = e;
                employers[e] = s;
                employers[eTemp] = -1;
                sue.push(eTemp);
            }else{
                sue.push(e);
            }


}
```

Poll has a runtime of O(log(n))
Assuming each match is created on first attempt for n employees and n employers the best runtime is n(log(n))

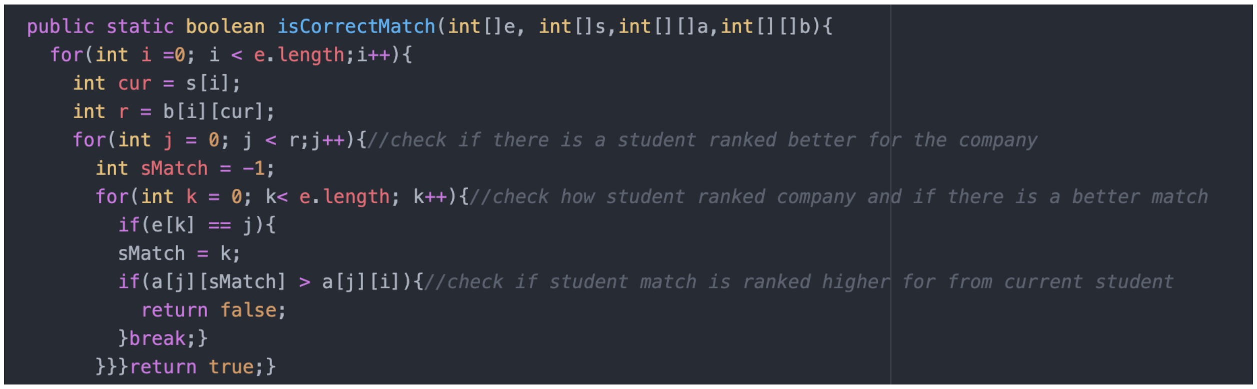
Worst run time:
n^2(log(n))
Until all matches are complete each Employers requests their most favored student, the student then says maybe, and it is saved until the student gets requested by a company, they favor the most, this is done until it cannot be optimised more. Therefore it can take up to n^2 time.

Stefan Todorovic 300118046

3.

a)

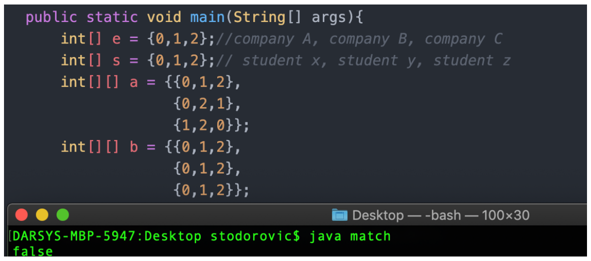
Code:



```java
public static boolean isCorrectMatch(int[]e, int[]s,int[][]a,int[][]b){
  for(int i =0; i < e.length;i++){
    int cur = s[i];
    int r = b[i][cur];
    for(int j = 0; j < r;j++){//check if there is a student ranked better for the company
      int sMatch = -1;
      for(int k = 0; k< e.length; k++){//check how student ranked company and if there is a better match
        if(e[k] == j){
        sMatch = k;
        if(a[j][sMatch] > a[j][i]){//check if student match is ranked higher for from current student
          return false;
        }break;}
      }}}return true;}
```

Verification:

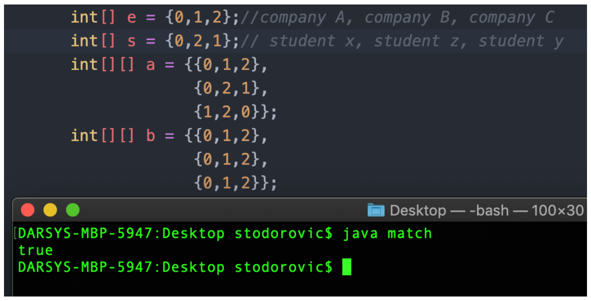
With non-stable match (example from pdf):



```java
public static void main(String[] args){
    int[] e = {0,1,2};//company A, company B, company C
    int[] s = {0,1,2};// student x, student y, student z
    int[][] a = {{0,1,2},
                 {0,2,1},
                 {1,2,0}};
    int[][] b = {{0,1,2},
                 {0,1,2},
                 {0,1,2}};
```

```
DARSYS-MBP-5947:Desktop stodorovic$ java match
false
```

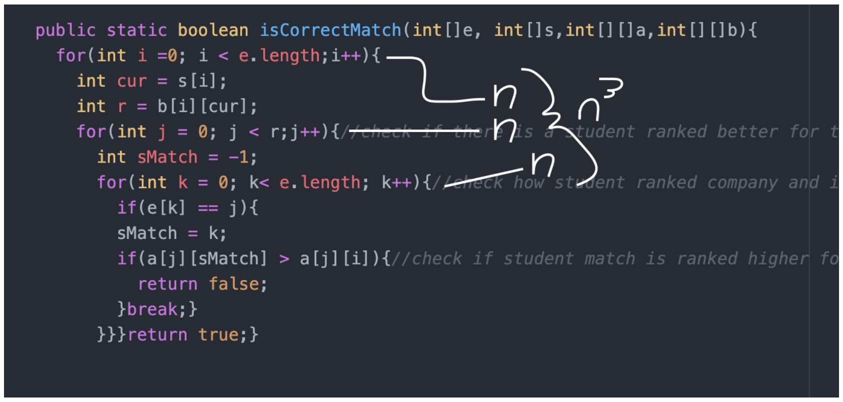
With stable match (example from pdf):



```java
    int[] e = {0,1,2};//company A, company B, company C
    int[] s = {0,2,1};// student x, student z, student y
    int[][] a = {{0,1,2},
                 {0,2,1},
                 {1,2,0}};
    int[][] b = {{0,1,2},
                 {0,1,2},
                 {0,1,2}};
```

```
DARSYS-MBP-5947:Desktop stodorovic$ java match
true
DARSYS-MBP-5947:Desktop stodorovic$
```

b)



```java
public static boolean isCorrectMatch(int[]e, int[]s,int[][]a,int[][]b){
  for(int i =0; i < e.length;i++){
    int cur = s[i];
    int r = b[i][cur];
    for(int j = 0; j < r;j++){//check if there is a student ranked better for t
      int sMatch = -1;
      for(int k = 0; k< e.length; k++){//check how student ranked company and i
        if(e[k] == j){
        sMatch = k;
        if(a[j][sMatch] > a[j][i]){//check if student match is ranked higher fo
          return false;
        }break;}
      }}}return true;}
```

n n n n^3

Worst runtime is O(n^3)