# T-106.1208 Basics of Programming Y (Python). Exam 24.10.2009

Write the following information clearly on top of each paper you submit: name of the course, date of the exam, your full name, student ID, the total number of papers you submit, and your signature.

Important instructions: Use indentations of the length of two squares in your code. If your indentations are not clear enough, you lose points. You do not have to write any comments in your code. You can assume that the input given by the user is correct, if it is not told in the problem that you should handle the incorrect input.

1. a) What is printed when the following Python program is executed? It is enough to give the correct output without any explanations. (2 p)

```
def main():
    temperature = 28.0
    if temperature >= 0.0:
        print "warm"
    elif temperature >= 25.0:
        print "hot"
    else:
        print "cold"

main()
```

b) Suggest an initial value for each of the variables `month` and `day` that will cause the following Python program print `Reduction 20 per cent`. It is enough to give one possible value for each variable. You do not have to list all possibilities. (2 p)

```
def main():
    month = ... # Three dots are replaced with a string
    day = ... # Three dots are replaced with an integer
    if month == "June" or month == "July":
        print "No reduction"
    else:
        if day > 10 and day < 16:
            print "Reduction 20 per cent"
        else:
            print "Reduction 40 per cent"

main()
```

c) What is printed when the following Python program is executed? It is enough to give the correct output without any explanations. (3 p)

```
def main():
    numbers = [4, -16, 21, 18, -3]
    result = 0
    for number in nubers:
        if number > 0:
            result = result + number
    print result

main()
```

d) Explain, in plain English using 1-2 sentences, the *purpose* of the following Python function `mystery1`. (Do not say *how* the code works. Instead say what the function would be used for.) You may assume that the parameters of function are two equally-long lists of integers. (4 p)

```
def mystery1(numbers1, numbers2):
    i = 0
    result = 0
    while i < len(numbers1):
        if numbers1[i] > numbers2[i]:
            result += numbers1[i]
        else:
            result += numbers2[i]
        i += 1
    return result
```

e) Explain, in plain English using 1-2 sentences, the *purpose* of the following Python function `mystery2`. (Do not say *how* the code works. Instead say what the function would be used for.) You may assume that the first parameter of the function is a list containing decimal numbers (floats) and the second and third parameters are decimal numbers. (4 p)

```
def mystery2(numbers, number1, number2):
    result = number2
    for item in numbers:
        if item < number1:
            result = result - item
    return result
```

f) Explain, in plain English using 1-2 sentences, the *purpose* of the following Python function `mystery3`. (Do not say *how* the code works. Instead say what the function would be used for.) You may assume that the parameter of the function is a list containing integers. (5 p)

```
def mystery3(numbers):
    i = 1
    while i < len(numbers):
        if numbers[i] != 2 * numbers[i-1]:
            return False
        i += 1
    return True
```

g) Explain, in plain English using 1-2 sentences, the *purpose* of the following Python function `mystery4`. (Do not say *how* the code works. Instead say what the function would be used for.) You may assume that the parameters of the function are two strings. Give an example of a practical purpose where the function could be used for. (5 p)

```
def mysteeri4(string1, string2):
    if len(string1) == 0 or len(string2) == 0:
        return "error"
    else:
        result = string1[0] + "." + string2[0] + "."
        return result
```

2. a) If you want to buy or sell shares (stocks), you can give a stock exchange order to your bank. If the bank can carry out your exchange oder, you must pay a brokerage fee to the bank. Assume that the brokerage fee of the bank is normally 0,25 per cent of the total value of the shares sold or bought. However, if the calculated fee is less than 10 euros, the brokerage fee is 10 euros. Write a Python program which asks the user to input the unit price of the share and the number of shares which were traded. (All shares have the same unit price.) After that, the program calculates and outputs the brokerage fee. (10 p.)

   b) A shop gives bonus points to its customers as follows: If the total value of items purchased in one day is less than `limit1` set by the shop, no bonus points are given. If the value is at least `limit1`, but less than `limit2` set by the shop, the customer earns 5 bonus points. If the value is at least `limit2`, the customer earns 10 bonus points. Write a Python function `count_bonus(purchases, limit1, limit2)`. The first parameter of the function is a list where each item is the total value of the items purchased in a certain day. The second and third parameters are the limits `limit1` and `limit2` set by the shop. The function must calculate and return the number of the bonus points the customer has earned. An example: if the list contains the numbers `12.0`, `30.0`, `20.0` and `12.0`, `limit1` is `15.0` and `limit2` is `25.0`, the function should return the value `15`. Write only the asked function. The main program or any other functions are not required. (20 p)

3. Write a Python program which asks the user to input a file name and a student id. The program reads this file. Each line contains at first the student id and after it other information belonging to this student (the name and the results of the student in a certain course, for example.) The program reads all lines and if one of the lines begins with the student id given by the user, the program outputs the whole line. If none of the lines begins with the given student id, the program tells the user that the student was not found. You may assume that the contents of the file is error-free: each line begins with the student id of length 6. You may also assume that the user gives a student id of length 6.

   However, your program has to be able to handle correctly (give a sensible error message) the error where the file is not found or it is not possible to read the file because of some other reason.

   **Hint**: After reading a line from the file, you can compare its six first characters with the student id given by the user. (20 p)

**The last problem is on the next page!**

4. Write a Python class FootballTeam to describe a football team in a league.

The class FootballTeam must have the following data attributes:

- __name name of the team
- __points the number of points the team has gathered in its matches.
- __goals_scored the number of goals the team has scored in its matches (total in all matches).
- __goals_conceded the number of goals the team has let its rivals score in its matches (total in all matches).

The class FootballTeam should have the following methods (if nothing is said about the return value of a method, it does not have to return anything):

- __init__(self, team_name) creates a new FootballTeam instance. The name of the team is given as a parameter. The points, scored goals and conceded goals of the new team are 0.
- get_name(self) returns the name of the team.
- get_points(self) returns the points of the team.
- count_goal_difference(self) calculates and returns the goal difference of the team (scored goals minus conceded goals).
- add_match(self, own_goals, rival_goals) adds information about one match, i.e. updates the points and goal statistics of the team according to the result of the match. The parameters of the method tell the number of goals the team has scored and the number of goals the rival team has scored in the match. The win gives 3 points, the draw 1 point and the loss 0 points. (You can use the goal information given as the parameters to decide if it was a win, a draw or a loss.) The method does not update the points and goal information of the rival team.
- is_better(self, anotherTeam) returns value True, if this team is better than the other team given as a parameter, and otherwise value False. When two teams are compared, the team which has higher points is better. If the points are equal, the team having higher goal difference is better. If the teams have both equal points and equal goal differences, the teams are equal. In this case, the method also returns value False.
- __str__(self) returns the string containing the name of the team, its points and its scored and conceded goals.

In addition, write a main function which creates two FootballTeam objects (instances) and calls add_match method to both of them. Then the main function has to call count_goal_difference for one team and output the goal difference. After that, the program must find out which team is better (by using is_better method) and output the name of the better team. If the teams are equal, the program may output either of the names. Finally, the program has to output the name, the points and scored and conceded goals of both teams. You may choose the names of the teams and the goals scored in the matches yourself. You do not have to ask the user to input them. (25 p)