



# **A PLUS**

## Institute Management System

### Technical Guide

Requirements of the system	3 - 5
Introduction	5 - 6
System overview	5 - 6
Important details	6 - 7
Behavior of the system	6 - 9
Functions	9 - 24
Text files	24 - 25
Contact details	25

# *Content*

# Requirements of the system

A plus institute hopes to fulfill these requirements from making a system.

- Register all current students for the system.
- Register new students.
- Keep a student records and access them when they required.
  - As a list
  - Individually
- Receive payments from both ALs and OLs and record them in a way that they can access that information later.
  - They are okay with record both classes' records in one file because this institute runs under a one teacher
  - But the class fee is different for two levels.
- To take each months records at the end of the month and keep it.
- Calculate the total received payments.

## **Introduction:**

'A Plus' institute is a well-known tuition class institution among A/L and O/L ICT students. This institute conducts classes for ICT under one teacher. And it hopes to develop a system for new student registration and charging monthly tuition fees from their students. The overall goals of the system are :

- Register new students to the institute/system.
- Collect student monthly fees
- Get student details.
- View student list
- Get payment record.
- Calculate monthly income

## **System overview:**

Main system interface will show these icons.

- Create a student account.
- Pay fees.
- View student profile.
- Students' records.
- Payment records.
- Institute staff options
- Exit

## **IMPORTANT DETAILS**

- Every month system user need to reset the payments and paid text files after taking its' backup.
- Index number is always given by the system when a new registration happens.
- This Institute is only for ICT subject under a one teacher. So the both AL and OL student fees calculated as one total income.

## **Behavior of the system**

### **Create a student account**

Staff member can input data directly. Insiute can use an application form to collect data.

#### **Giving an index number to them.**

When the system turning on staff member have to give the last taken index number to the system. So then system can automatically provide the next available index number when a new student registration is happening. To give this staff member can check the last number of students.txt file.

When this system is going to introduce all the students have to enter their information and get registered to this new system. It can be done by the student by entering their information.

And when a new student comes to the institute that new student also have to register to this institute. Registration fees will be charge at the counter after registration. If the entered index number is not equal to given number system will ask to redo the registration. So that index numbers won't make any mistakes.

Above details will record in a text file called students. Student record can be taken by that too.

## **Pay fees**

In every new month students have to pay fees.

Student payments details are on a “payment” text files. And the class fees values for A/L and O/L are different.

So when a student comes to pay fees system asked for a confirmation. For this student can show her/his last month card to the staff member. If staff member got the confirmation then system will show how much that student have to pay or not it will ask for reason.

Basically there can be 2 reasons. Lost card or not paid for last month. If it is a lost card system will show a message to the user and guide he student to a staff member and then they can check her/his paid status by last month payment record which they already took a backup at the end of that month.

If the student didn't pay system will ask for full payment for 2 month.

And when student made a payment system will show a message that the payment is successful and record it on both payments and paid text files.

- Paid text file shows who paid for his month. And staff have to take a backup at the end of the each month. They can rename it with the month name and keep it for future use.
- Payment text file only record the amount of payments. When teacher wants to know the total income they can use this file to calculate it.

## **View student profile**

Teacher or staff members can access student details is any emergency situation occurred. As for an example,

- If student get sick suddenly

- Teacher wants to send a message to students

Those details will be useful. When user enters the index number of a student it will show all the details recorded in students.txt for that index number.

### **Students' records**

By this system can access all the data in students.txt and show them as a list.

### **Payment records**

If the institute needs a record for this month

- who paid the fees

System can provide that as list accessing paid.txt.

### **Institute staff options**

By this option staff can know the total income of the month. For this it needs a password to access this option. Staff can check it at the end of the month and record it. In here system take data from payments.txt and transfer it to an array and after that it calculate the total income.

And here it reminds staff to reset those text files in beginning of the every month.

### **Exit**

Exit from the system.

Except this exit method user can come back to the main menu after completing the task at each window.

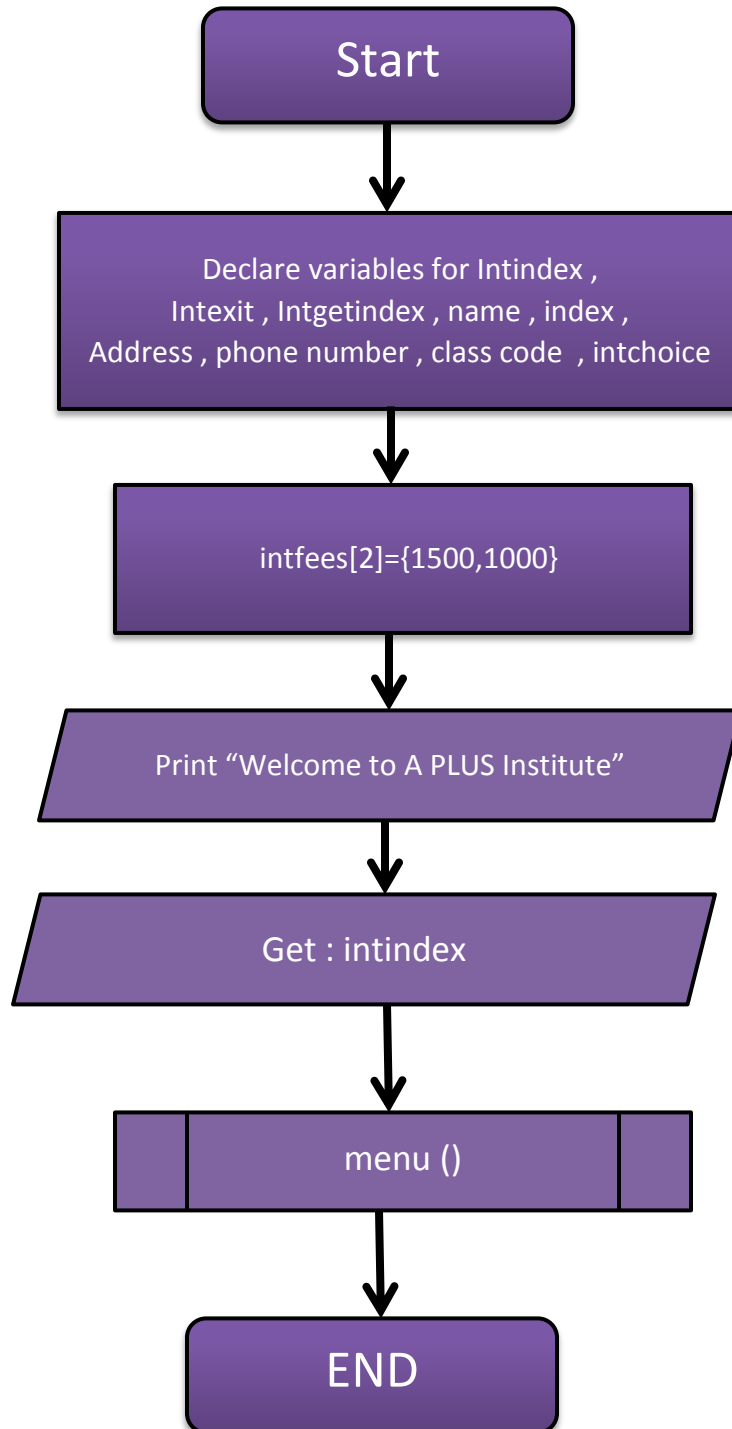


# FUNCTIONS

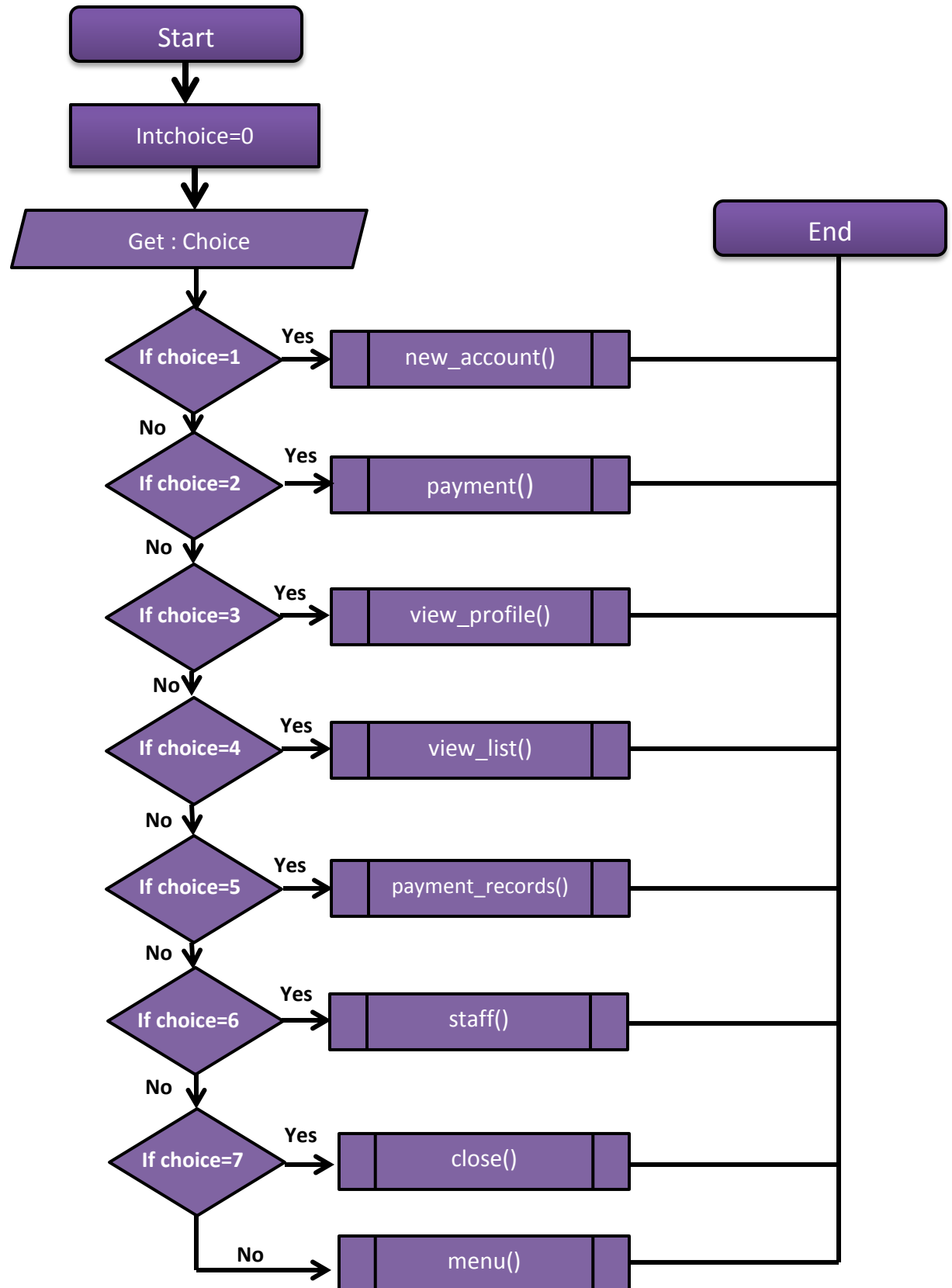
### Used functions

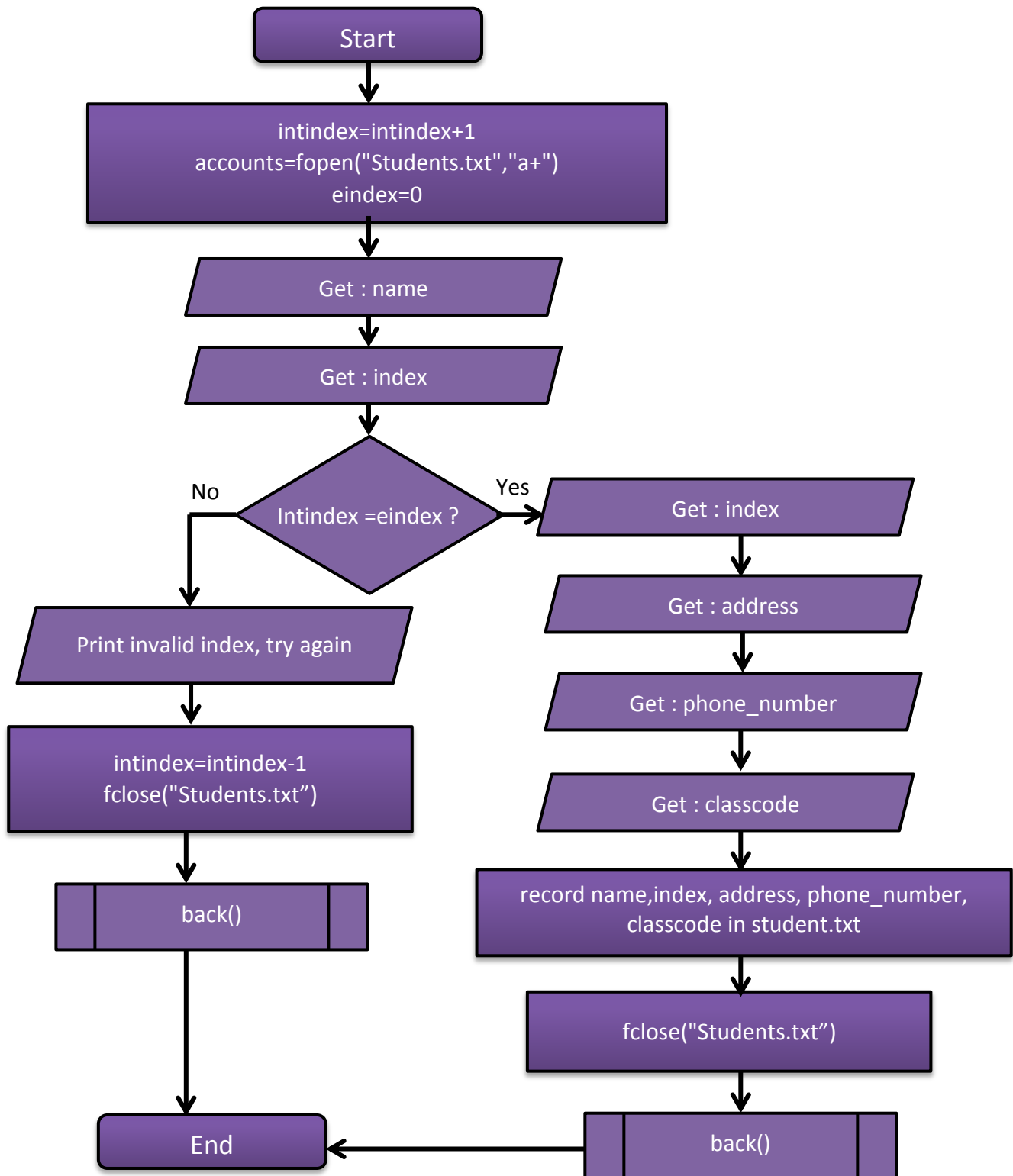
Function name	Return type
menu()	void
new_account()	void
view_profile()	void
payment()	void
view_list()	void
payment_records()	void
staff()	void
income()	void
back()	void
close()	void
invalid()	void
functionpaid()	void

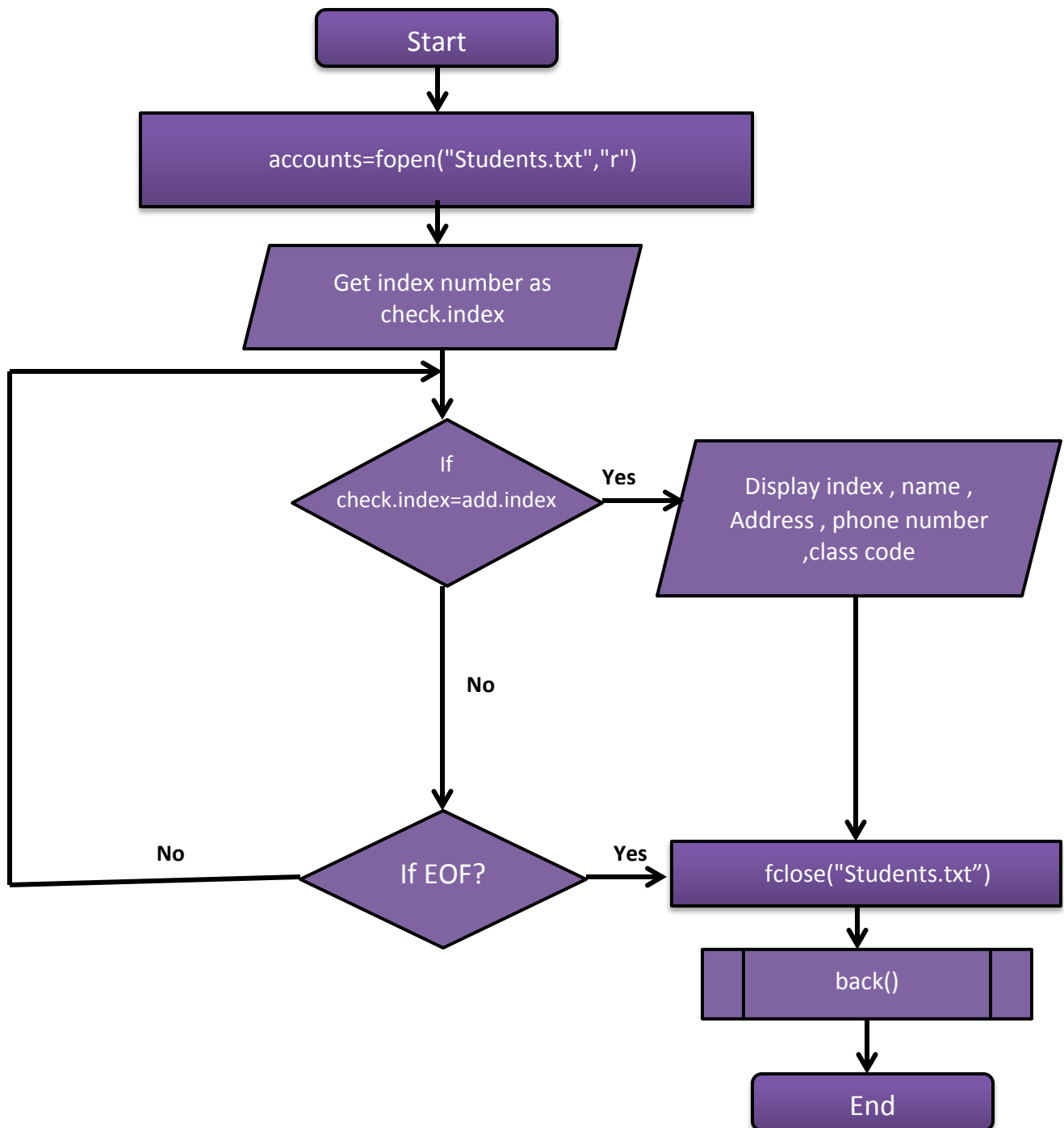
## Main function



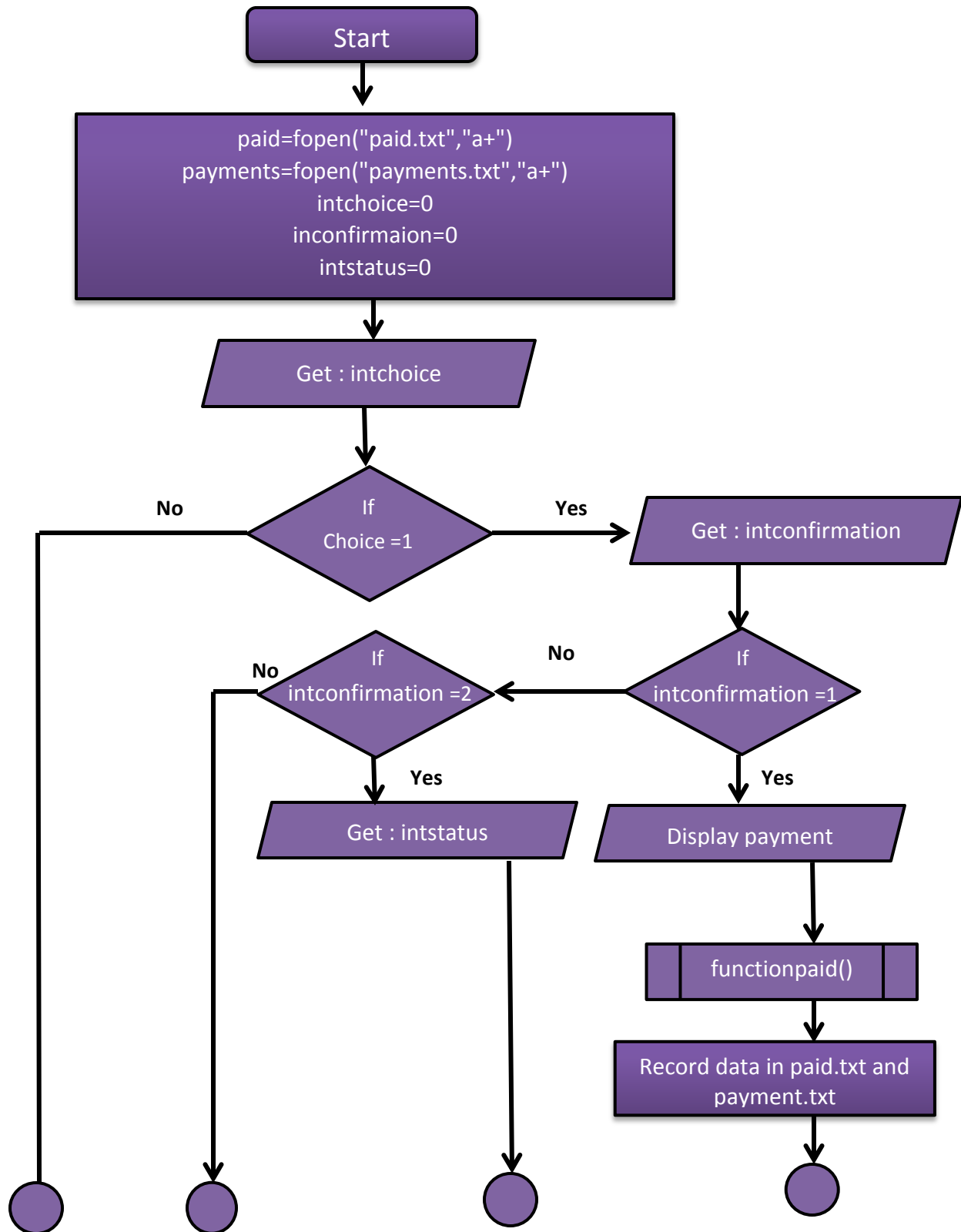
## Menu function

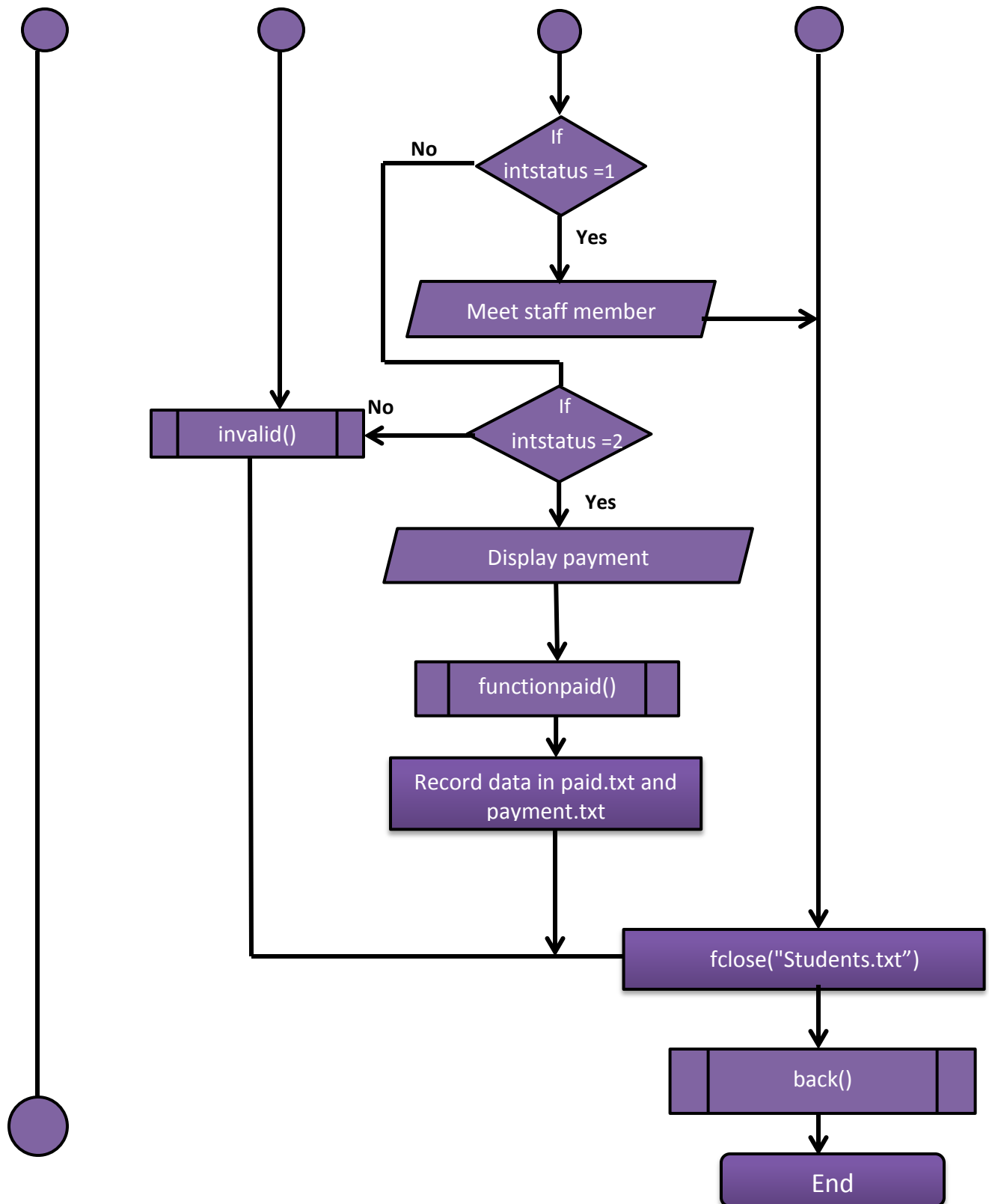


**new\_account function**

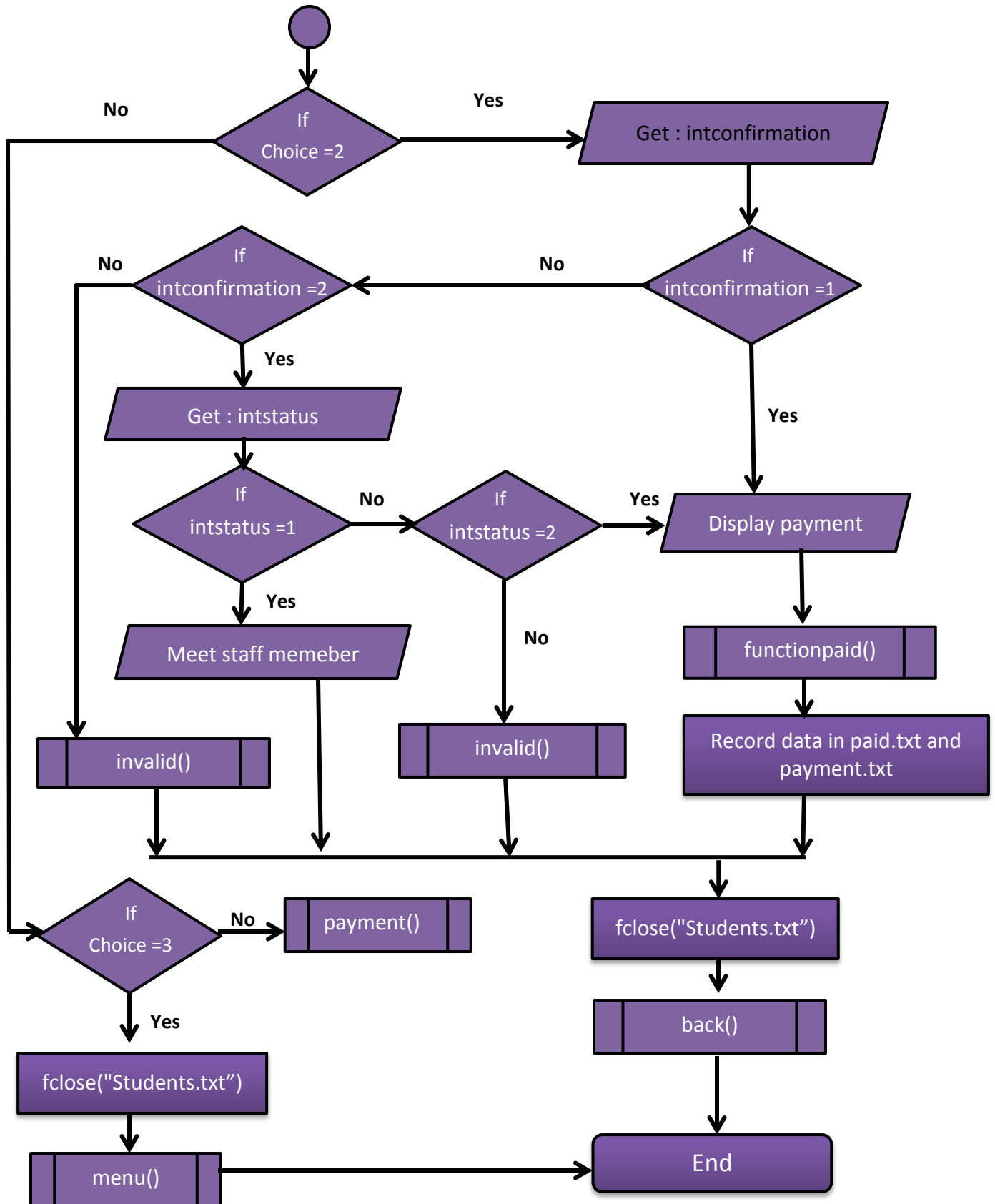
**view\_profile function**

## Payment function

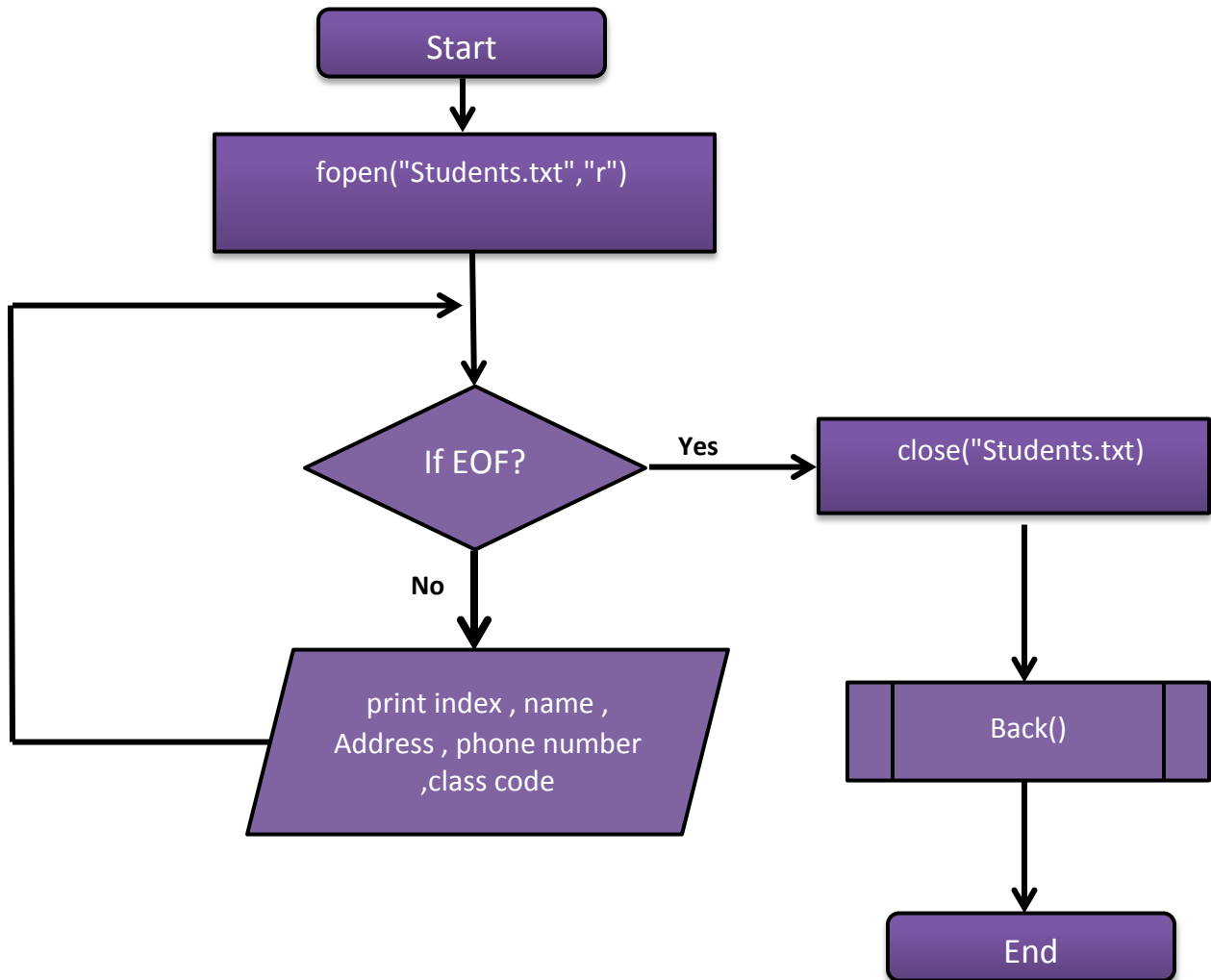




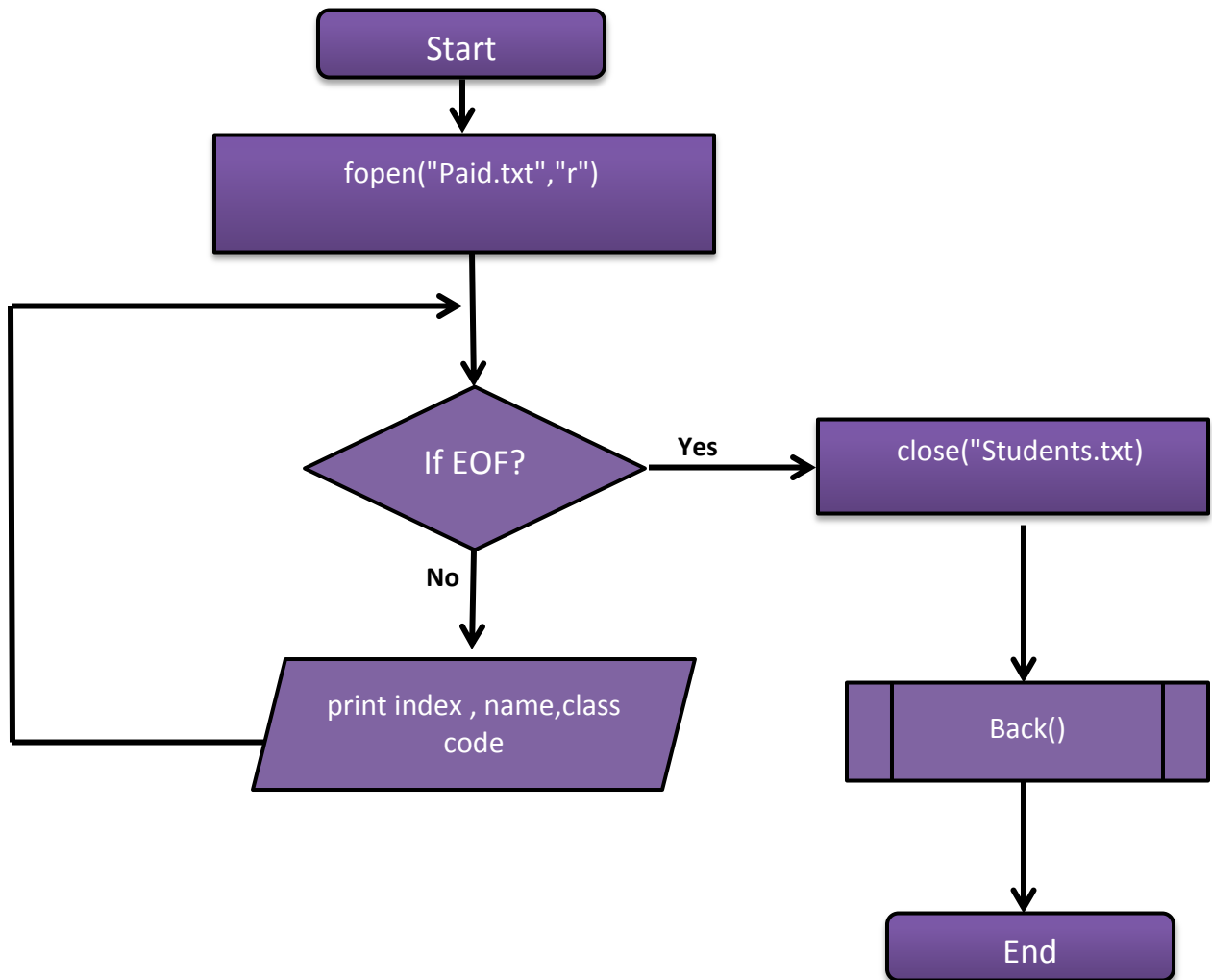




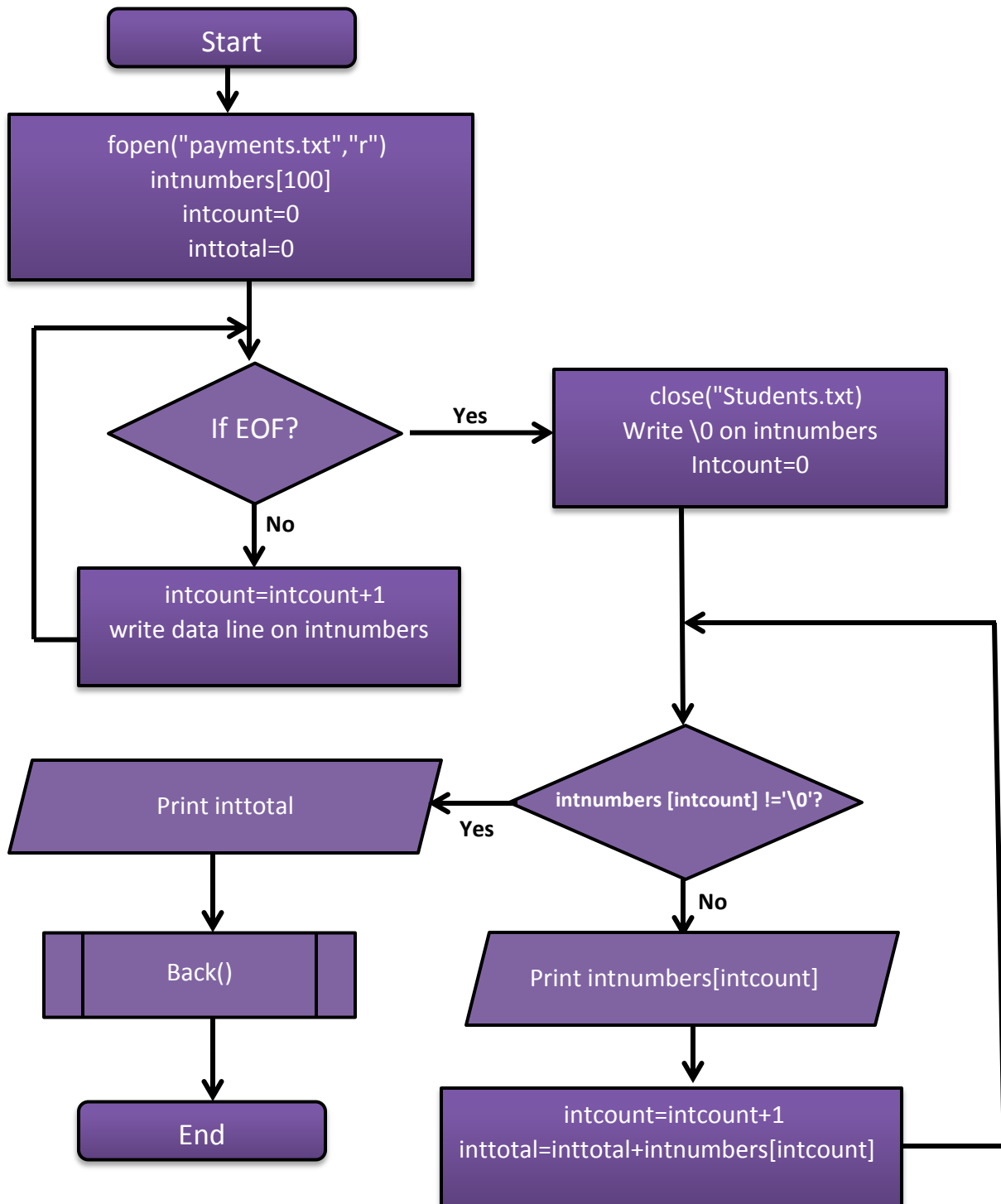
## View list function



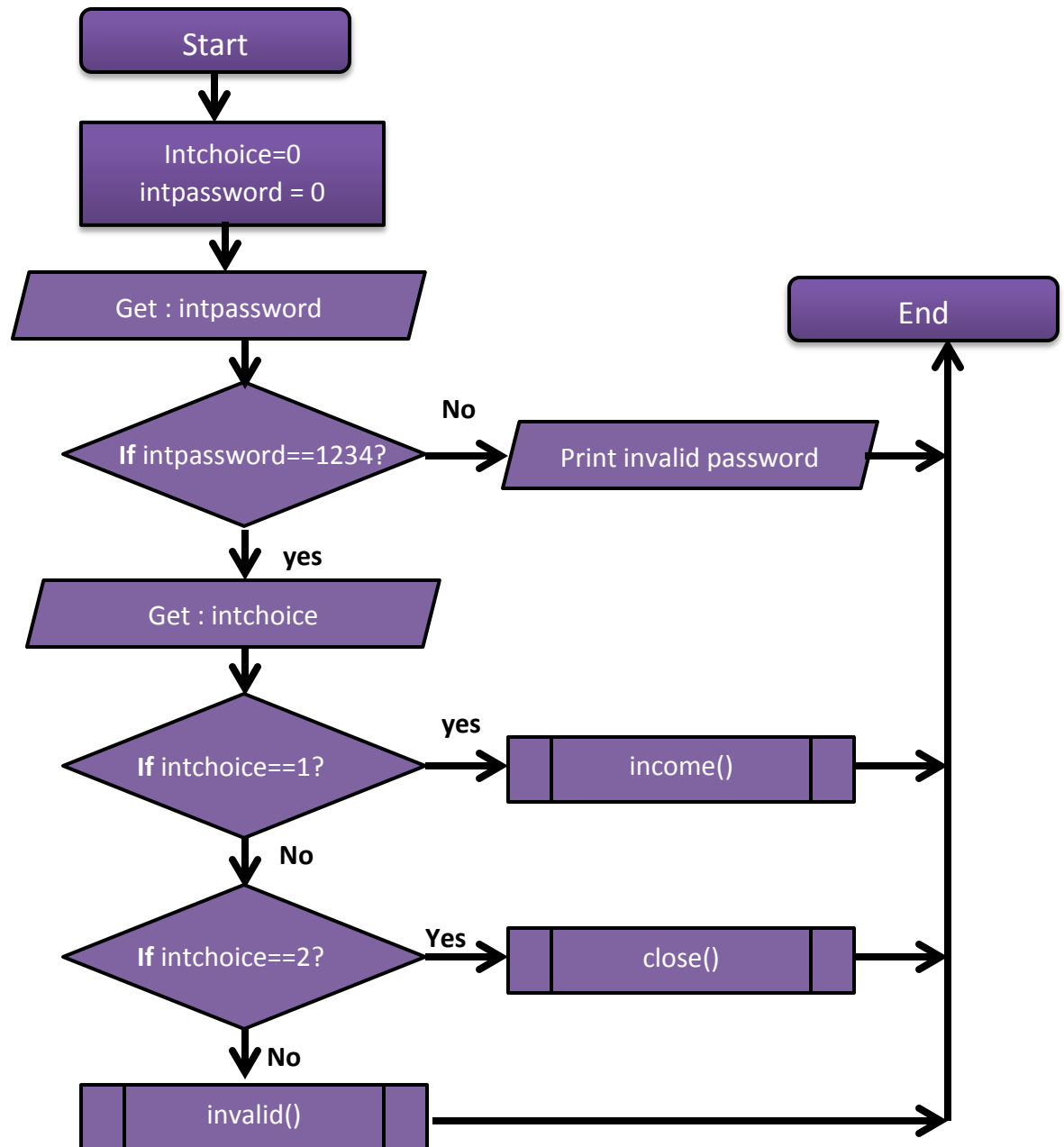
## Payment record function



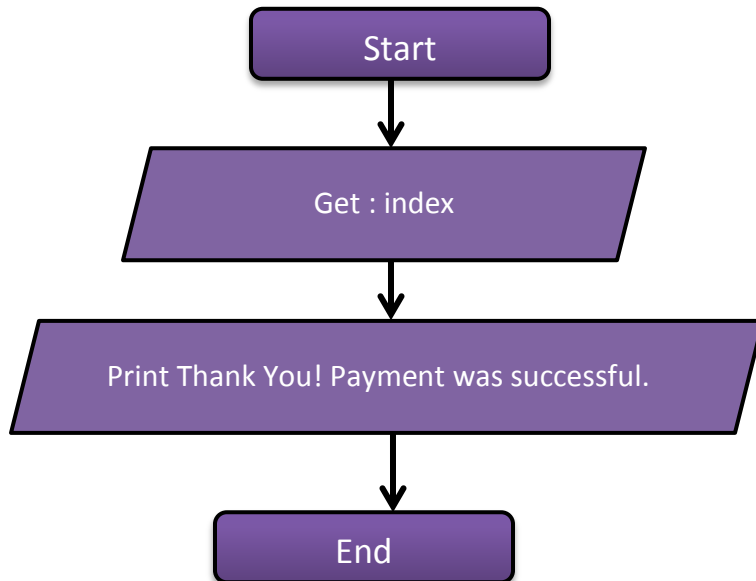
## Income function



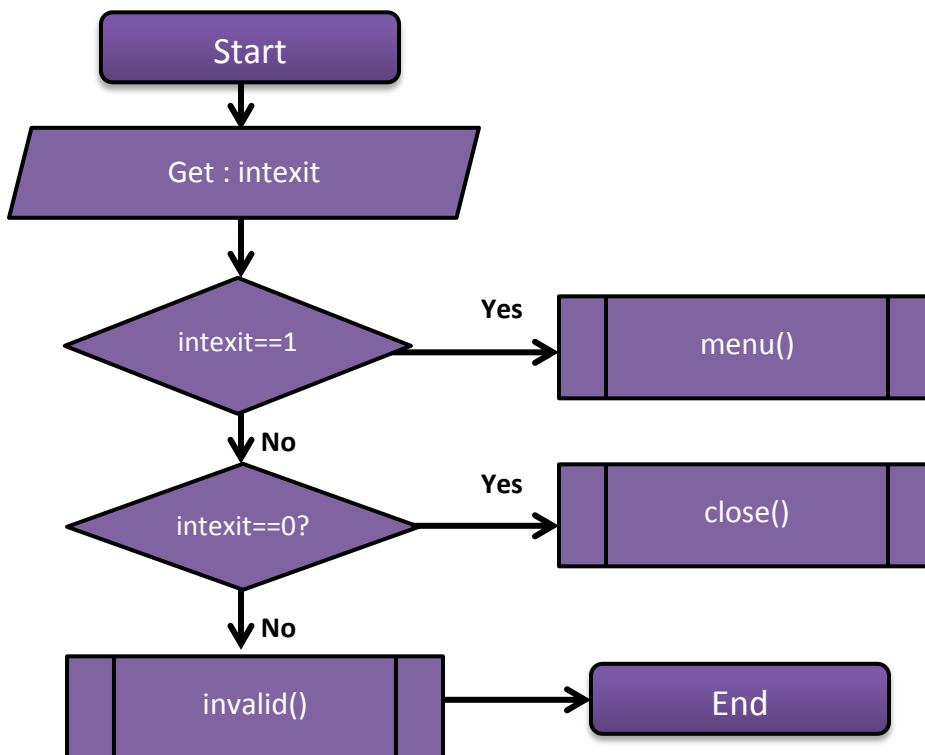
## Staff function



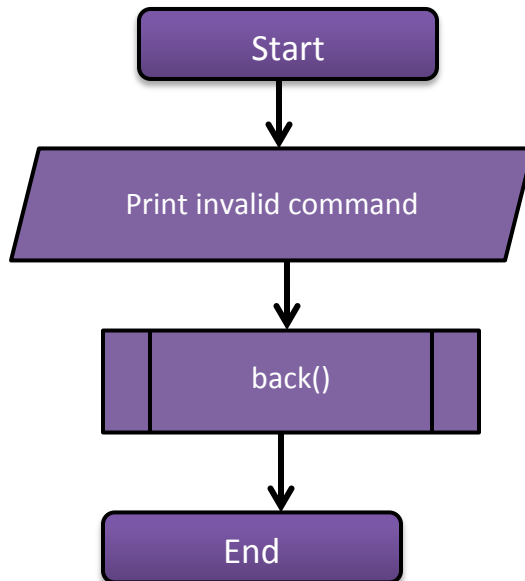
### Function paid function



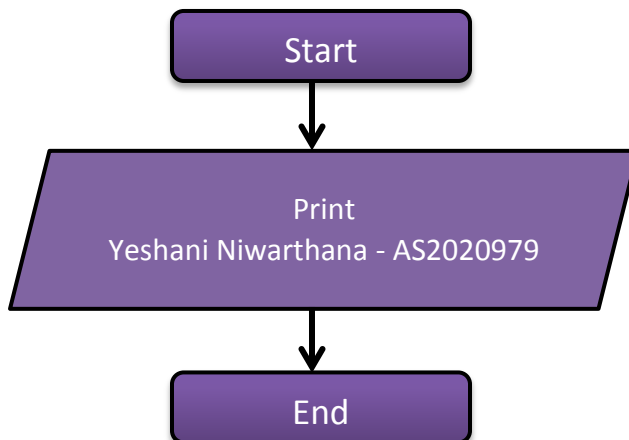
### Back Funcion




## Invalid function



## Close function



## Text files

 Students.txt

 Payments.txt

 Paid.txt



A PLUS Institute Management System

Version 1.0

Created by K.D.Y.Niwarthana

AS2020979

For any changes or modifications contact us – 0713554846

Gmail – [as2020979@sci.sjp.ac.lk](mailto:as2020979@sci.sjp.ac.lk)

For more details

