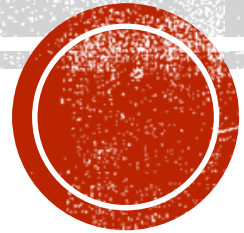
Hands-on Exercises

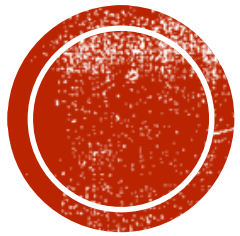
(Strings and Dynamic Allocation)

IT 1201



The problems are found inside my folder in Drive T: (teacherfiles)

Instructions:



1. Create a folder with the date today as its name. January 7, 2016
2. Place all your outputs today in the folder you just created.

Save as: faveFood.c

```
#include<stdio.h>
#define N 5
typedef struct{
 char fname[30], lname[30];
}name;

typedef struct{
 name stud_name;
 char favoriteFood[30];
 char gender;   /* F for female and M for male*/
 int age;
}student;
```

```
void input(student *s, int x);
void display(student *s, int x);
int maleAppetite(student *s, int x, char FOOD[]);

int main(void)
{
 _________ stud[N];
 char FOOD[10];
 int i;
 _____________ /* func call to input() */
 printf("\n\n");
 _____________ /* func call to display() */
 printf("Enter food to search: ");
 ___________________________________
 _____________ /* func call to maleAppetite() inside a printf */
 return 0;
}
```

Given the structure definition, finish the program e by creating the following functions:

Save as: faveFood.c

Given the structure definition, finish the program by creating the following functions:

- **input()-** asks the user to input the contents of the array of students
- **maleAppetite()-** that accepts a string FOOD and an array of students. The function will count and return how many of the male population has the FOOD as their favorite food.
- **display()** - that displays the contents of the array in this format:

  Lastname, Firstname Gender FavoriteFood Age

Save as: books.c

Based on the structure definition on the right, complete the main() function and write the function definitions stated below given their function prototypes.

The main() function asks the user to input how many books are to be inserted and calls the necessary functions to do the insertion if there is still room/space for the books. Otherwise, no insertion is made if it is already full. All books in the array are then displayed.

main() will also ask the user to input a book price and display 3 books of the same price.

The main() function is found on the next slide.

```
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>
#define SIZE 1000

typedef struct {
  char author[30], title[30], publisher[30];
}PrimeInfoType;

typedef struct {
  PrimeInfoType BkInfo;
  int YearPub;
  float BookPrice;
}Book;

typedef struct{
  Book bks[SIZE];
  int ctr; /* contains the actual # of books in bks */
}ArrBooks;
```

```
Book inputBook(void);
int addBooks(Book bk, ArrBooks *A);
void displayBooks(ArrBooks A);
Book* searchBooks(float price, ArrBooks A);

int main(void)
{
 ArrBooks newList;
 Book b, *list;
 int x, num, catch;
 newList.ctr = 0; /* no books yet */

 printf("\nEnter no. of books to input: ");
 ______________________

 for(x = 0; x < num; x++)
 {
  b = inputBook();
  catch = _________   /* addBooks() func call */
  switch(______)
  {
  case 1: printf("\nSuccessfully added!\n");
          break;
  case 2: printf("\nOverflow of books!");
          break;
  }
 }

/* continue at the right */
```

```
 _____________     /* displayBooks() func call */
 printf("\n\nEnter book price: ");
 scanf("%f", &price);
 ___________________ /* searchList() func. Call */

 for(x=0; x<3; x++){   /* prints only the first 3 search results*/
    _________________ /* print format is the same with displayBooks() */
 }

 getch();
 return 0;
}
```

Save as: books.c

Code the function definitions of the following :

- **Book inputBook(void);** — Asks the user to input and return the details of ***ONE*** book.
- **int addBooks(Book bk, ArrBooks *A);** — adds ***ONE Book bk*** to the array of books in ***ArrBooks *A*** if there is still space ***(refer to ctr)*** and return 1 if successful.
- **void displayBooks(ArrBooks A);** — displays all the books in the following format below:

  1997 Harry Potter and the Sorcerer's Stone *created by* ***J.K. Rowling*** *published by* ***Bloomsburry***

- **Book *searchBooks(float price, ArrBooks A);** - returns an array of books with the same price based on the given parameter.