"Computer Principles for Programmers(CPR101/BTP105)"Final Project

Introduction

Students, working in teams of 4 to 5, will create a console application to demonstrate various operations with null-terminated C strings.

Application Modules

Module#1–Fundamentals(`fundamentals.h`,`fundamentals.c`) Module

#2 – Manipulating (`manipulating.h, manipulating.c`) Module #3 –

Converting (`converting.h, converting.c`)

Module#4–Tokenizing(`tokenizing.h`,`tokenizing.c`)

Each module is divided into three blocks:

Fundamentals->	**1)Indexing**	**2)Measuring**	**3)Copying**
Manipulating->	**1)Concatenation**	**2)Comparison**	**3)Search**
Converting->	**1)Convertingto`int`**	**2)Convertingto`double`**	**3)Convertingto`long`**
Tokenizing->	**1)TokenizingWords**	**2)TokenizingPhrases**	**3)TokenizingSentences**

Each team has three "junior" programmers responsible for modules1-3, one "senior" programmer (team leader) responsible for module 4, synchronizing and integrating modules into the main application, and one tester. If there are 4 students in a team, the team leader does the testing as well. On project completion all the team members will have the same mark.

Application Versions and Deadlines

In Version#1 students implement first block s of code only, in Version#2 – the first and the second blocks, in Version #3 – all three blocks. Each version takes one week approximately to complete.

Project Details

Students will not develop modules (we are not IPC144).All the modules versions will be provided by the teacheras `.png` (graphics) files. So, students will have to ***type*** code, ***comment***, ***compile***, ***test***, ***stage*** and ***commit*** versions using `git` program; they will ***communicate*** and ***collaborate*** through MS Teams. The programmers will send modules to the tester.The tester will develop test cases as an Excel spread sheet. After the modules testing is done, the team leader combines all the sources in one app. Only the team leader will submits the source files, the test cases and screenshots (if needed) through the Blackboard.

Rubrics

A teacher may mark the final project with a C/C+ if version#1 was completed, a B/B+ if version#2 was completed, and an A/A+ if students team reached the final version#3.The final project mark may also depend on the quality of students' comments, efficiency of their communications monitored by the teacher, ability to meet deadlines and on application results.

Appendix A Standard Library C Functions used by Modules

Fundamentals Module

strlen() //length
strcpy() //copy

Manipulating Module

strcat() //concatenation
strcmp() //comparison
strstr() //search

Converting Module

atoi() //string to int
atof() //string to double
atol() //string to long

Tokenizing Module

strtok() //tokenizing

Appendix B -Students' Responsibilities, Versions, Marks, and Tools Used

	Version #1 (required) Grade:"C/C+" Tools:cl/gcc	**Version #2(optional) Grade:"B/B+" Tools:cl/gcc, git**	**Version #3 (optional) Grade:"A/A+" Tools:cl/gcc, git**
Junior Programmer#1 Fundamentals Module	Indexing	Add measuring	Add copying
Junior Programmer#2 Manipulating Module	Concatenating	Add comparing	Add search
Junior Programmer#3 Converting Module	Converting to int	Add converting to double	Add converting to Long
Senior Programmer/Team Leader Tokenizing Module	Tokenizing words	Add tokenizing phrases	Add tokenizing sentences
Tester	Spread sheetv1	Spread sheetv2	Spread sheetv3

Appendix C-Deliverables and Deadlines

Version#1-"C/C+"grade(week12) Due: April1, Tuesday,2025

1. fundamentals.h
2. fundamentals.c
3. manipulating.h
4. manipulating.c
5. converting.h
6. converting.c
7. tokenizing.h
8. tokenizing.c
9. main.c
10. All the modules should have comments of the code
11. "module"-test-cases.xls(individual modules test cases, here module should replace with your module's name)
12. "module"-testing.txt(individual modules console text captured as text showing test case inputs and outputs)
13. test_cases_v1.xls(version1 final test cases)
14. Updated Project plain.xlsx

Version#2-"B/B+"grade(week13) Due: April 8, Tuesday,2025

1. fundamentals.c
2. git_status_log1_screenshot.txt
3. manipulating.c
4. git_status_log2_screenshot.txt
5. converting.c
6. git_status_log3_screenshot.txt
7. tokenizing.c
8. git_status_log4_screenshot.txt
9. main.c
10. All the modules should have comments of the code
11. "module"-test-cases.xls(individual modules test cases, here module should replace with your module's name)
12. "module"-testing.txt(individual modules console text captured as text showing test case inputs and outputs)
13. test_cases_v2.xls(version2 final test cases)
14. Updated project plan.xlsx

Version#3- "A/A+"grade(week14) Due:April14,Monday 2025

1. fundamentals.c
2. git_ status_log1_screenshot.txt
3. manipulating.c
4. git_ status_log2_screenshot.txt
5. converting.c
6. git_ status_log3_screenshot.txt
7. tokenizing.c
8. git_ status_log4_screenshot.txt
9. main.c
10. All the modules should have comments of the code
11. "module"-test-cases.xls(individual modules test cases, here module should replace with your module's name)
12. "module"-testing.txt(individual modules console text captured as text showing test case inputs and outputs)
13. test_cases_v3.xls(all the versions final test cases in one file)
14. Final project plan.xlsx

Compiler: You should compile your program by using gcc compiler or visual studio . According to your platform you should use the appropriate one.

Windows user: You should download a Linux emulator called Cygwin (www.Cygwin.com) to run your program using gcc compiler. Please watch the following

video to know how to install Cygwin and configure the path to your command line.

https://youtu.be/DaiJoWr5fH0 Mac

Mac user: Watch the following video

https://youtu.be/we2Oc4WQ7FM

Ms teams channel:

All you files should be uploaded on MSteams, in your private channel. Only the final version of your project will be submitted by your team leader using the blackboard. All the project related communications should be done by using the ms teams.