CS 442/542 Project Check List

Integer Variables and Expressions
15 Points

- Integer Literals, Variables

- Variables must be declared before they are used (all declarations before other statements)

- Assignment Statement

- Arithmetic Operators

  - +, - (binary subtraction), *, /, %, - (unary minus)

  - ^ (exponentiation) You can assume the exponent is an expression whose value >= 0

- All operators except ^ use the precedence and associativity rules of C. ^ is right associative and has lower precedence than unary minus and higher precedence than all the other operators. Parentheses are used to change order of operations.

Boolean variables and Expressions
15 Points

- Boolean Literals, Variables

- Variables must be declared before they are used (all declarations before other statements)

- Assignment Statement

- Relational Operators

    - <, <=, ==, >=, >, !=

- Boolean Operators

    - !, &&, ||

- All operators except use the precedence and associativity rules of C. Parentheses are used to change order of operations. Use C definition of true (non-zero) and false (0)

I/O
25 Points

- read(comma delimited list of variables)

  - read(x, y, z)

  - 1 (true) and 0 (false) can be input for boolean variables

- print(comma delimited list of expressions);

  - print results on the current line

  - print **a space** between values

  - print true or false for boolean values

  - a newline is not printed

  - print(2*x, x^3*(z*w), abc/xyz, c)

- printlines(expression)

  - print expression number of newlines

  - printlines(2)

- printspaces(expression)

  - printspaces(3*n)

- printString("Literal String")

Control Structures
25 Points

- if statements including optional else

- while loops

- Use C syntax and the C definition of true (non-zero) and false (zero).

Arrays
25 points

- Single Dimension Arrays

  - Size of the array is declared at compile time

- Use C syntax

- Subscripts must be able to use constants and expressions. For example nums[2*i] = z * nums[j-3*k]

2D Arrays
20 Points

- Size of the array is declared at compile time

- Use C syntax

- Subscripts must be able to use constants and expressions.

Functions 1
15 Points

- void, int and boolean parameterless functions

- Recursion

-

Functions 2
30 Points

- void, int and boolean functions

- Parameters: Pass by value for integers

- Parameters: Pass by reference for arrays

- Local Variables

- Recursion

- Pass by reference for integers (bonus 5 points)

- Pass by value for arrays (bonus 5 points)

Project

- You must do the items in the order shown on the previous slides. For example before you do arrays you must have integer and boolean expressions, I/O and the control structures implemented. New features must work with the old features. For example when you implement arrays you must be able to read and print array elements.

- I will give you some test programs written in C. You can translate these programs to your language (usually only the I/O will need to be changed). You should also create your own test programs

-

Project Submission

- "On time" demonstrations will be on Friday, Saturday or Monday December 9, 10, 12

- 5 bonus points if you demonstrate **on or before** December 7

- -10 points late penalty for demonstrating on December 13 or 14

- To submit your project you will upload one zip file to Canvas. The file will contain your source code, a README file listing the parts of the project you completed, the test programs you ran, the assembly code you generated and files containing the results (copy of the terminal window output of the test programs). There must be three files for each test program (the source, the assembly and the results). Name the files in a consistent pattern. For example t1.c, t1.asm, t1Result.txt