Assignment 7: Converting Bases

EC602 Design by Software

Fall 2021

Contents

1 Introduction

1.1 Assignment Goals

The assignment goals are to help you learn about

- arithmetic operators
- proper choice of integer types
- number systems and bases

1.2 Group Size

The maximum group size is 2 students.

1.3 Due Date

The assignment is due 2021-11-15 at 23:59:59

1.4 Assignment Value

This assignment is worth 5 points.

1.5 Late policy

Late assignments will be accepted until the beginning of the lecture immediately following the due date, or for 48 hours, whichever is less.

If the *natural grade* on the assignment is g, the number of hours late is h, and the number of hours between the assignment due time and the next class is H, the new grade will be

```
(h > H) ? 0 : g * (1- h/(2*H))
```

If the same assignment is submitted ontime *and* late, the grade for that component will be the maximum of the ontime submission grade and the scaled late submission grade.

1.6 Submission Link

You can submit here: convertbase hw7 submit link

1.7 Background on bases

Every number, when written or represented in text, uses a *base representation.* For example, the number 255 means

$$2 \times 10^2 + 5 \times 10^1 + 5 \times 10^0$$

because we are using base-10 representation.

When a number y is written in base n as

$$d_m d_{m-1} ... d_2 d_1 d_0$$

then y can be calculated as

$$y = \sum_{i=0}^{i=m} d_i n^i$$

and in addition, we restrict the values of the digits as follows:

The first digit satisfies $0 < d_m < n$ and all others have $0 \leq d_i < n$

2 Convertbase

Write a program `convertbase.cpp` which converts a number, represented as a string in one base, to a new string representing that number in a new base. You may assume that the number can be stored in an `int` without overflow.

The character to represent a digit with value `digitvalue` is the ASCII character `digitvalue+'0'`.

Note that this means that the conventional use of a-f for bases like 16 is not supported by `convertbase`.

The program should expect three command line arguments

- a string representing the number to convert
- the base that the preceeding string is represented in
- the base that the number should be converted to.

The values of the original base and the target base will always be in the range 2 to 200 inclusive.

The program should print the new representation to the standard output and then exit.

2.1 Restrictions

You may only include the libraries `iostream` and `string`. No other includes are permitted.

2.2 Checker

There is a checker available here:

hw7_convertbase_check.py