The Heritage Computer Challenge
1999
Heritage High School
Newport News, Virginia
C++ Division

Welcome

Welcome to the Heritage Computer Challenge for 1999! You are to be commended for taking the time and making the effort to be here today. Have a great time and may all your programming efforts be successful!

~Mr. Charles F. Monroe, Contest Director

Instructions

The problems for this contest appear on the following pages, listed in order of difficulty. The maximum number of points you can earn is indicated under the title to each problem.

Problems are designed in the format used by The Great Computer Challenge, held annually each Spring at Old Dominion University. Some of these problems were actually used at the Great Computer Challenge in previous years.

Solutions should be saved as a project on your personal drive K **in a folder whose name is IDENTICAL to the project file name** (minus the extension). Source file names are up to you, as long as their extension is **.cpp**.

Three Sailors and a Monkey
(10 points)

Save in folder named: Sailors

Three sailors, shipwrecked with a monkey on a desert island, have gathered on one day a pile of coconuts that are to be divided early the next day. Sometime during the night, one sailor arises, divides the pile into three equal parts, and finds one coconut left over, which he gives to the monkey. He then hides his share, and returns the remaining coconuts to a single pile. Later during the same night, each of the other two sailors arises separately and repeats the performance of the first sailor. In the morning all three sailors arise, divide the pile into three equal shares, and find one coconut left over, which they give to the monkey.

Write a program in Pascal that will compute how many coconuts were in the original pile. Since there is more than one correct answer, the program should consider all coconut piles in the range of 1 to 1000. The output should be displayed on the screen and consist of the following:

a. The number of coconuts in the original pile.
b. The number of coconuts after each sailor removes a third.

One correct answer is 79 and may be used to check the correctness of the program. Output for this pile could look like the following:

```
Coconuts in the original pile      79
Coconuts after the first sailor    52
Coconuts after the second sailor   34
Coconuts after the third sailor    22
```

Pyramid of Letters
(20 points)

Save in folder named: Pyramid

The Pyramid is a structure found in many cultures. This shape is often associated with supernatural power. You are asked to write a program that accepts a single character from "A" through "Z" and produces an output in the shape of a pyramid composed of the letters up to and including the letter that was input. The top letter in the pyramid should be an "A" and on each level, the next letter in the alphabet should fall between the letter that was introduced in the level above it.

EXAMPLE: (bolded values denote user input)

```
 Please enter the letter of choice:

E

Your pyramid is as follows:


           A
          ABA
         ABCBA
        ABCDCBA
       ABCDEDCBA


Are there more letters? Enter Yes or No
```

The Heritage Computer Challenge *1999*

C++ Division

Palindromes
(20 points)

Save in folder named: Palindromes

A palindrome is a word, phrase, verse, paragraph, etc., which reads the same forwards or backwards (excluding punctuation, spacing, and capitalization). For example, the following are palindromes:

Madam, I'm Adam.
Poor Dan is in a droop.
Now Sir, a war is never even. Sir, a war is won.
Sue Zues.
Evade Dave.

Write a program which takes an arbitrary list of alphabetic characters and determines if it is a palindrome or not.

Input for each palindrome will not exceed 80 characters. Note that blanks (spaces), capitalization, and punctuation do not affect the determination of a palindrome. Numeric characters will not occur in the input. Your program should consider only one palindrome at a time, i.e., for each run of the program, there is only one palindrome to read.

Output should be PALINDROME if the expression is a palindrome, or NOT A PALINDROME if it is not.

Example:

Palinedrome Test
Not now, no strap parts on Won Ton.
PALINDROME

The Heritage Computer Challenge *1999*

C++ Division

Perfect Numbers
(30 points)

Save in folder named: PerfectNumbers

The Greeks began an examination of numerology by classifying all positive integers as either perfect, abundant, or deficient. This classificatin scheme is based on the factors (even divisors) of the number. If the sum of all of the factors of a number (excluding the number itself) equals the number then it is said to be "perfect". For example, the factors of 6 are 1, 2, 3, and 6. Therefore, the number 6 is a perfect number. The total of the factors of 6 (excluding the number itself, in this case 6) is 1+2+3 = 6.

An abundant number is one in which this sum of factors (excluding the number itself) is greater than the number. An example of an abundant number is 12, because the sum of the factors of 12 is greater than 12. ex. 1+2+3+4+6 = 16 which is greater than 12. All numbers that are neither perfect nor abundant are deficient.

Write a program that prompts the user to enter a positive integer (allow integer values between 1 and 500). The program should at this point display the original number, the factors in that number and whether the num ber is perfect, abundant, or deficient.

EXAMPLE: (bolded values denote user input)

```
Please enter a positive integer:  6
The factors of 6 are: 1, 2, 3, 6
The number 6 is perfect

Please enter a positive integer: 12
The factors of 12 are: 1, 2, 3, 4, 6, 12
The number 12 is abundant

Please enter a positive integer: 333
The factors of 333 are: 1, 3, 9, 37, 111
The number 333 is deficient
```