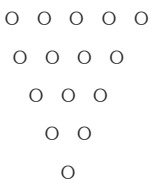
Functions

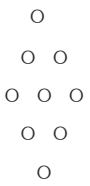
- HomeWork,Answers
- Reference Parameters Section 5.3
- Review Questions
- Exam October 22.
  (Chapters 1,2. Sections 3.1,3.2,3.3,3.4,3.6,4.1 4.2,4.4,4.5,4.7,4.8,5.1,5.2,5.3)

HomeWork

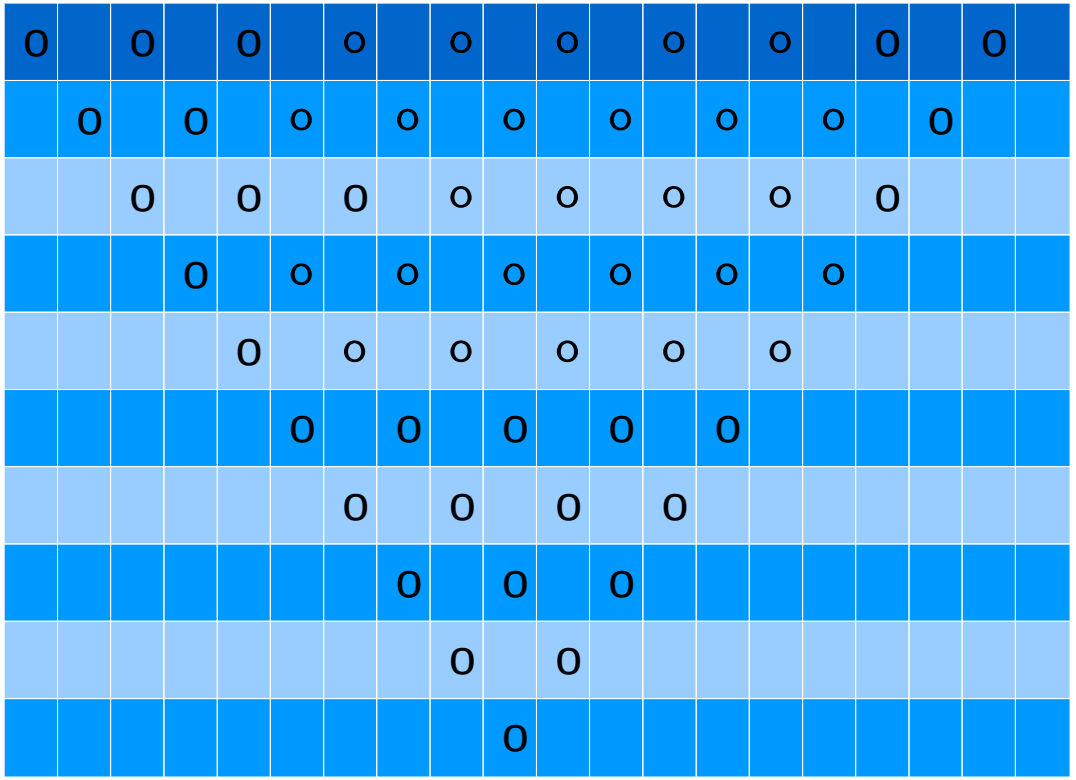
- Write a function to print out a triangle with n lines that looks like this



- Combine this function together with the function pretty_triangle1 in a program that will ask the user for a number of lines and print out a set of n dots the pattern below. The program is due in class on Oct 20. Email the program to me at ruttan@cs.kent.edu with the subject line 10061 submission Homework 3.



Planning your function



Functions with more than one answer

- Read numbers from a file and find the maximum number and its position in the file and the minimum number and its position in the file;
- Want to write a function but it will need to return 4 answers.
- Use a special type of parameter that uses a memory location in the program calling (invoking) the functions. Called reference parameters.

Reference Parameters

- Parameter which contain a & after the data type and before the paramter name are called reference parameter. Examples
  - void in_out(double& a, int &b, char & c);
  - Int calc_function(float & xxx, double & a);
- Unlike nonreference parameter, the memory that holds the parameter value is located in the calling program. This means that a change made to a reference parameter in the function changes the value of the argument in the classing program

Reference Parameter Example

```
intM minMax(ifstream infile, double& minval, int& minPos,double& maxval,int&
   minPos ){
   int count=0;
   double val;
   if (infile.fail()) return count;
   infile >> val; count=1;
   minval=val; maxval=val;minPos=count;maxPos=count;
   while (!infile.fail()){
       infile>>val; count=count+1;
       if (val > maxval) {
            maxval =val;maxPos=count;
       }
      if (val < minval) {
            minval =val;minPos=count;
       }

   }

   if (infile.eof) return 0; else return count;

}
```

- Complete Program