# Sample LaTeX document

## Question 1

Suggested document structure: give your answers to each question in sections. Answers to subquestions in subsections, etc.

### Part (a)

Here is an example algorithm:

```
procedure CheckNumbers(A,B)                          ▷ A and B are two lists of integers
    count = 0
    for i = 1,...,n do
        for j = i...m do
            if A[i] ≥ B[j] then
                count = count +1
                break
            end if
        end for
    end for
end procedure
```

It will be inserted exactly where it appears in the document. This is not recommended because it is better to group the entire algorithm together and "float" the group to somewhere where it will all fit: this will make things more readable. As an example here is a second, much longer, algorithm (see Algorithm 1).

Observe that the `algorithm` environment also adds nice visual structures around your algorithm.

You should use `\ref` and `\label` to refer to your algorithm in the text so that the reader can find it. Note that if you use `\ref` and `\label` you need to compile your latex document twice: the first time grabs all the labels (and puts them into the .aux file) and the second time it can insert the numbers into the file. Otherwise your references look like this: **??**.

LaTeX will sort out page layout for you in a sensible/recommended way (e.g. page breaks are inserted "intelligently"), so you shouldn't tinker too much with things, just let LaTeX do its own thing.

## Question 2

The `enumerate` package lets you do tailored enumerates:

I First

II Second

III Third

This can be useful if you want to do a simple algorithm:

**Algorithm 1** A longer algorithm

```
procedure LongerCheckNumbers(A,B)
                                          ▷ A and B are two lists of integers
    count = 0
    for i = 1,...,n do
        for j = i...m do
            if A[i] ≥ B[j] then
                count = count +1
                break
            end if
        end for
    end for
    count = 0
    for i = 1,...,n do
        for j = i...m do
            if A[i] ≥ B[j] then
                count = count +1
                break
            end if
        end for
    end for
    count = 0
    for i = 1,...,n do
        for j = i...m do
            if A[i] ≥ B[j] then
                count = count +1
                break
            end if
        end for
    end for
    count = 0
    for i = 1,...,n do
        for j = i...m do
            if A[i] ≥ B[j] then
                count = count +1
                break
            end if
        end for
    end for
end procedure
```

Step 1. Make a sandwich

Step 2. Eat the sandwich

If you want to break an enumerate with text and then resume, use `\setcounter{enumi}{..}` to start the counter from somewhere specific. For example

Step 0. Buy sandwich ingredients

## Question 3

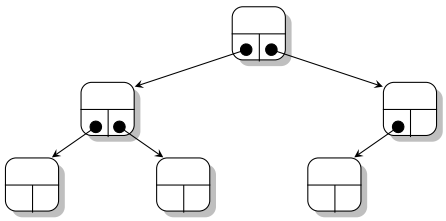
The `listings` package lets you list code verbatim, with some syntax highlighting:

```
\documentclass{article}

\usepackage{listings}

% recursion alert: stack overflow
```

A list of supported languages can be found here.

## Question 4

`TikZ` is a very useful and powerful package for drawing pictures directly in LaTeX. The documentation and library of examples are quite extensive, but the results can be pretty:



## Question 5

There is almost always a LaTeX package for your typesetting needs. Most packages are well documented and many of the popular ones have copious amounts of examples of their capabilities. Google is your friend.