The idea is to allow entry into a critical section only if it is "our turn" or "either turn." The "either" value avoids the requirement that processes must alternate entry into their critical sections. Initially, `turn` = `either`. Does this work?

8. [5 pt] Show that, if the `wait` and `signal` operations are not executed atomically, then mutual exclusion may be violated.

9. [5 pt] Write a procedure using `cobegin/coend`, or `fork` and `join` primitives (but not both), to compute the dot product $\sum_{i=1}^{n} a_i \cdot b_i$ of two vectors in parallel.

4. [4 pt] There are the following two files in your directory:

```
- file1
" file 2 "
```

   You want to delete the first file and edit the second. What commands will you use? You must account for the space characters in both the files.

5. [5 pt] Suppose two processes P_1 and P_2 each contain the following high level language commands in a loop:

```
numres = numres + 1
if numres = 50 then numres = 0
```

   Suppose also that both share the variable `numres` and that both are active. Initial value of `numres` is 1. Is it possible that `numres` will never be reset to zero? Why? Will your reasoning hold if there was only one CPU executing both processes?

6. [5 pt] Show that in the bakery algorithm, the following property holds: If P_i is in its critical section and P_k ($k \neq i$) has already chosen its `number`$[k] \neq 0$, then `(number[i],i)` $<$ `(number[k], k)`.

7. [6 pt] Consider the following implementation of critical section problem for two processes:

```
type
   turntype = (one, two, either);
var
   turn :  turntype; /* Shared variable */

process p_i;
var
   ours, theirs :  turntype
      /* In process one, ours = one and theirs = two */
      /* In process two, ours = two and theirs = one */
repeat
   while (turn = theirs) do;
   turn = ours;
      critical_section;
   if other process is suspended then
      turn = theirs;
      resume other process;
   else
      turn = either;
until false;
```

CS 376 **Operating Systems** Test 1
Name: Maximum Points: 46

Important: This is a closed book test. You should not have anything except the test sheet on your desk. *Do not log into the computer during the test.* Any calculations and rough work should be done on the back side of the test pages only.

1. [6 pt] Consider the following applications. Would each be best categorized as batch, interactive, or real-time?

   (a) word processing

   (b) program that prints monthly checking and savings accounts summaries

   (c) language compiler used by university students

   (d) airline reservation system

   (e) aircraft navigational system

   (f) hotel reservation system

   (g) software that monitors fission in a nuclear power plant

   (h) program that computes weekly averages for members of a bowling league

   (i) architects CAD (Computer Aided Design) system for creating drawings

   (j) library checkout system

   (k) telephone switching system

   (l) building security system

2. [5 pt] You have the following command in your `.cshrc` file.

   ```
   set noclobber
   ```

   What is the effect of this command? How can you nullify the effect in your shellscript (without using the `-f` option of the `C` shell)?

3. [5 pt] In Unix, a set of files with the same extension are required to be renamed such that the extension is changed. For example, we may have a set of files like `haar.c`, `ht.c`, `gt.c`, `h.c`, and `g.c`. There may be other files in the directory such as `*.h` files, man pages, and `Makefile`. However, our `C` compiler requires the file extension to be `.cc` instead of `.c`. Provide a command or shell script that will change all the `.c` files to `.cc` files. The result of this command will be to rename the above example files to `haar.cc`, `ht.cc`, `gt.cc`, `h.cc`, and `g.cc` while leaving the other files as they are. Your commands should work for a change from any extension to any other extension.