1. (5 pt.)

```
#include <sys/types.h>
#include <signal.h>
#include <stdio.h>
#include<stdlib.h>
#include <unistd.h>

void terminated (int sig)
{
        if (sig == SIGUSR1)
        {
               printf("Mom said I need study! Bye! I need go! \n");
                _exit(0);// child send SIGCHLD when it is terminated
        }
        else if (sig == SIGCHLD)
        {
               printf("Please don't disturb my child studying! \n");
                _exit(0);
        }
}

int main()
{
   pid_t pid;
   int i, count;

   if ((pid = fork()) <0)
   {
      perror ("fork error");
      exit (1);
   }
   if (pid >0)/* parent process */
   {
        count =0;
        while (1)
        {
                signal (SIGCHLD, terminated);
                printf("Son! It is time to study\n", count+1);
          sleep(1);
          if (count==10)
                 kill(pid, SIGUSR1); //send SIGUSR1 to child
                count++;
        }
     }


   else /* child process */
   {
     while (1)
     {
          printf("Mom! I want play forever!\n");
          signal (SIGUSR1, terminated);
          sleep(1);
     }
    }
  }
```

2. (5 pt.)

```
#include <pthread.h>
#include <stdio.h>

void *thread1(void *);
void *thread2(void *);
void *thread3(void *);
int main (int argc, char *argv[])
{
      if (argc ==1)
      {
           perror("argument number error");
           exit (1);
      }
      int num = atoi(argv[1]);
      int rc;
      pthread_t tid1, tid2, tid3;
    void *tret1, *tret2, *tret3;

      rc=pthread_create(&tid1, NULL, thread1, (void *) num);
      rc =pthread_join(tid1, &tret1);
      rc=pthread_create(&tid2, NULL, thread2, (void *) tret1);
      rc =pthread_join(tid2, &tret2);
      rc=pthread_create(&tid3, NULL, thread3, (void *) tret2);
      rc =pthread_join(tid3, &tret3);
      printf ("the final result of three theads's calcuation
is %d\n", (int) tret3);
      exit (0);
}
void *thread1(void *arg)
{
      int num = (int)arg;
      num = num +10;
    return ((void *) num);
}

void *thread2(void *arg)
{
      int num = (int)arg;
      num = num +20;
      return ((void *) num);
}

void *thread3(void *arg)
{
      int num = (int)arg;
      num = num +30;
      return ((void *) num);
}
```