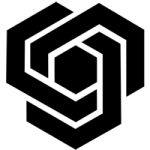
Технически Университет – София



Системно програмиране

**Курсова работа**

Изготвил:Проверил:

Иван Александров Боев /121213011/ /В. Поляков/

гр.41

**Задание:**

Да се реализира приложение, което "изтегля" торент по зададен адрес. Задайте директория, където искате този торент да бъде изтеглен и запишете във файл името на изтегленото съдържание (напр. "GTA V"), неговия размер и адреса, от който е дошъл. Реализирайте две програми, като първата е клиент (съдържа името на потребителя и адреса на файла, реализира записа на торента на PC). Тя изпраща тези данни до сървъра (друга програма), а той от своя страна записва във файл всички изтегляния, които потребителите са направили, и при ново стартиране на програмата, ги възстановява в свой контейнер. Ако потребител иска да направи отново дадена заявка, то трябва да се изпише, че тя вече е извършена и не може да бъде направена отново. Демонстрирайте работа с повече от един клиенти.

За реализация на проекта се използва езика C.

**Сървър:**

Отваря се сървър на порт 8000, като чрез fork() има възможност за повече едонвременно свързани клиенти (ограничени до 5 в случая). Когато се свърже нов клиент, в сървъра се принти неговия адрес и потребителското име което той избира. Когато клиентът изпрати информация за това какъв файл е „свалил“, тази информация се запазва в текстов файл.

**Клиент:**

Свързва се със сървъра на порт 8000, като при свързването е подканен да си избере потребителско име. След това има възможност да „свали“ един от няколкото файла, като информация за файла се изпраща на сървъра.

GitHub: <https://github.com/ivanboev/SPr>

Source code:

Server:

#include<stdio.h>  
#include<string.h>  
#include<stdlib.h>  
#include<sys/types.h>  
#include<sys/socket.h>  
#include<netinet/in.h>  
#include<arpa/inet.h>  
#include<unistd.h>  
#include<sys/stat.h>  
#include<sys/sendfile.h>  
  
#define PORT 8000  
#define MAXSZ 100  
int main()  
{  
 int sockfd;  
 int newsockfd;  
  
 struct sockaddr\_in serverAddress;  
 struct sockaddr\_in clientAddress;  
  
 int n;  
 char msg[MAXSZ];  
 int clientAddressLength;  
 int pid;  
  
 //create socket  
 sockfd=socket(AF\_INET,SOCK\_STREAM,0);  
 //initialize the socket addresses  
 memset(&serverAddress,0,sizeof(serverAddress));  
 serverAddress.sin\_family=AF\_INET;  
 serverAddress.sin\_addr.s\_addr=htonl(INADDR\_ANY);  
 serverAddress.sin\_port=htons(PORT);  
 printf("Server waiting for clients\n");  
  
 bind(sockfd,(struct sockaddr \*)&serverAddress, sizeof(serverAddress));  
  
 listen(sockfd,5);  
  
 while(1)  
 {  
 clientAddressLength=sizeof(clientAddress);  
 newsockfd=accept(sockfd,(struct sockaddr\*)&clientAddress,&clientAddressLength);  
 printf("connected to client: %s\n",inet\_ntoa(clientAddress.sin\_addr));  
  
   
 pid=fork();  
 if(pid==0)  
 {  
 n = recv(newsockfd,msg,MAXSZ,0);  
 if(n==0)  
 {  
 close(newsockfd);  
 break;  
 }  
 FILE \*fp;  
 fp=fopen("savedRequests.txt", "a");   
 printf("New client connected: %s", msg);  
   
 while(1)  
 {  
   
 fprintf(fp,"%s",msg);  
 n=recv(newsockfd,msg,MAXSZ,0);  
 fprintf(fp,"%s",msg);  
 printf("File info saved");  
 if(n==0)  
 {  
 close(newsockfd);  
 break;  
 }  
   
 }  
 exit(0);  
 }  
 else  
 {  
 close(newsockfd);  
 }  
 }  
  
 return 0;  
}  
  
  
Client:

#include<stdio.h>  
#include<sys/types.h>  
#include<sys/socket.h>/  
#include<string.h>  
#include<stdlib.h>  
#include<netinet/in.h>  
#include<arpa/inet.h>  
#include<unistd.h>  
  
#define PORT 8000  
#define SERVER\_IP "127.0.0.1"  
#define MAXSZ 100  
int main()  
{  
 int sockfd;  
  
 struct sockaddr\_in serverAddress;  
  
 int n;  
 char msg1[MAXSZ];  
 char msg2[MAXSZ];  
 FILE \*fp;  
 int choice;  
  
 //create socket  
 sockfd=socket(AF\_INET,SOCK\_STREAM,0);  
 //initialize the socket addresses  
 memset(&serverAddress,0,sizeof(serverAddress));  
 serverAddress.sin\_family=AF\_INET;  
 serverAddress.sin\_addr.s\_addr=inet\_addr(SERVER\_IP);  
 serverAddress.sin\_port=htons(PORT);  
  
 connect(sockfd,(struct sockaddr \*)&serverAddress,sizeof(serverAddress));  
  
 printf("\nChoose your username:\n");  
 fgets(msg1,MAXSZ,stdin);  
 if(msg1[0]=='#'){  
 return(-1);  
 }  
 n=strlen(msg1)+1;  
 send(sockfd,msg1,n,0);  
   
 while(1)  
 {  
 {  
 printf("Download file: 1:GTAV.txt, 2:WoW.txt, 3:DOTA2.txt\n");  
 scanf("%d", &choice);  
 char info[1024];  
 switch(choice)  
 {  
 case 1:  
 fp = fopen("GTAV.txt", "r");  
 fgets(info,1024,fp);  
 send(sockfd, "GTAV.txt", 8, 0);  
 n = strlen(info) + 1;  
 send(sockfd,info,n,0);  
 break;  
 case 2:  
 fp = fopen("WoW.txt", "r");  
 fgets(info,1024,fp);  
 send(sockfd, "WoW.txt", 7, 0);  
 n = strlen(info) + 1;  
 send(sockfd,info,n,0);  
 break;  
 case 3:  
 fp = fopen("DOTA2.txt", "r");  
 fgets(info,1024,fp);  
 send(sockfd, "DOTA2.txt", 9, 0);  
 n = strlen(info) + 1;  
 send(sockfd,info,n,0);  
 break;  
 }  
  
 n=recv(sockfd,msg2,MAXSZ,0);  
  
 printf("Received message from server::%s\n",msg2);  
 }  
}  
 return 0;  
}