**객체지향프로그래밍 LAB #07**

**<기초문제>\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. 아래의 프로그램을 작성하시오.(/\*구현\*/ 부분을 채울 것, 표의 상단: 소스코드, 하단 : 실행결과)

|  |
| --- |
| #include <iostream>  #include <string> //string객체 사용  #include <fstream> //ifstream ofstream  #include <iomanip> // setw  using namespace std;  int main() {  string s1 = "mouse";  cout << s1 << endl;  cout << /\*구현\*/ << ", " << /\*구현\*/ << endl << endl; //s1의 length, size 출력  cout << s1.empty() << endl;  s1.clear();// s1 = "";  cout << s1.empty() << endl;  s1 = "Good";  s1 = s1 + "-bye";  // Good-bye (string)  // 01234567 (index)  cout << s1 << endl;  cout << /\*구현\*/ << ", " << /\*구현\*/ << endl; // 두가지 방법으로 index 4 값 출력  cout << /\*구현\*/ << endl; // s1과 Good-bye 비교 (bool 값 출력)  cout << /\*구현\*/ << endl; // s1과 good-bye 비교 (bool 값 출력)  cout << /\*구현\*/ << endl << endl; // s1이 “z”보다 크거나 같은지 비교 (bool 값 출력)  cout << /\*구현\*/ << endl; // s1의 index 5부터 3글자 출력  cout << /\*구현\*/ << endl; // s1의 index 2부터 2글자 출력  cout << /\*구현\*/ << endl; // od 위치 출력  cout << s1.find("od", 5) << endl;  int od\_index = s1.find("od");  cout << s1.find("od", od\_index + 2);  cout << (s1.find("korea") == string::npos) << endl;  return 0;  } |
|  |
| #include <iostream>  #include <string> //string객체 사용  #include <fstream> //ifstream ofstream  #include <iomanip> // setw  using namespace std;  int main() {  string s1 = "mouse";  cout << s1 << endl;  cout << s1.length() << ", " << s1.size() << endl << endl; //s1의 length, size 출력  cout << s1.empty() << endl;  s1.clear();// s1 = "";  cout << s1.empty() << endl;  s1 = "Good";  s1 = s1 + "-bye";  // Good-bye (string)  // 01234567 (index)  cout << s1 << endl;  cout << s1[4] << ", " << s1.at(4) << endl; // 두가지 방법으로 index 4 값 출력  cout << (s1=="Good-bye") << endl; // s1과 Good-bye 비교 (bool 값 출력)  cout << (s1=="good-bye") << endl; // s1과 good-bye 비교 (bool 값 출력)  cout << (s1>"z") << endl << endl; // s1이 “z”보다 크거나 같은지 비교 (bool 값 출력)  cout << s1.substr(5,3) << endl; // s1의 index 5부터 3글자 출력  cout << s1.substr(2, 2) << endl; // s1의 index 2부터 2글자 출력  cout << s1.find("od") << endl; // od 위치 출력  cout << s1.find("od", 5) << endl;//5번째 인덱스부터 "od"를 찾아봐  int od\_index = s1.find("od");  cout << s1.find("od", od\_index + 2);  cout << (s1.find("korea") == string::npos) << endl;  return 0;  } |
|  |

2. 아래의 프로그램을 작성하시오.(/\*구현\*/ 부분을 채울 것)

|  |
| --- |
| #include <iostream>  #include <string> //string객체 사용  #include <fstream> //ifstream ofstream  #include <iomanip> // setw  using namespace std;  int main() {  ofstream fout // processor->file 저장  /\*구현\*/ // example.txt 열기  string s2 = "Objective Oriented Programming";  fout << s2 << endl;  fout << "Random Variables" << endl;  fout << "Linear Algebra" << endl;  /\*구현\*/ // fout 닫기  ifstream fin;  string s1;  /\*구현\*/ // example.txt 열기  if (!fin) {  cout << "Error, no such file exists" << endl;  exit(100);  }    while (/\*구현\*/) { // line by line으로 example.txt에서 읽어와서 출력  cout << s1 << endl;  }    /\*  //한번에 실행되지 않기 때문에 주석처리 해놓음  while (1) { // 띄어쓰기 단위로 example.txt에서 읽어와서 출력  /\*구현\*/  }  \*/  fin.close();  } |
|  |
| #include <iostream>  #include <string> //string객체 사용  #include <fstream> //ifstream ofstream  #include <iomanip> // setw  using namespace std;  int main() {  ofstream fout; // processor->file 저장  fout.open("example.txt");  //fout.open("C:\\Users\\KYJ\\Desktop\\3학년 1학기\\객프\\실습 과제\\example.txt"); // example.txt 열기  //fout.open("..\\..\\example.txt"); // 상대 경로.  string s2 = "Objective Oriented Programming";  fout << s2 << endl;  fout << "Random Variables" << endl;  fout << "Linear Algebra" << endl;  fout.close(); // fout 닫기  ifstream fin;  string s1;  fin.open("example.txt"); // example.txt 열기  if (!fin) {  cout << "Error, no such file exists" << endl;  exit(100);  }  while (getline(fin, s1)) { // line by line으로 example.txt에서 읽어와서 출력  cout << s1 << endl;  }  //한번에 실행되지 않기 때문에 주석처리 해놓음  //while (1) {  // fin >> s1;  // if (!fin) break;  // cout << s1 << endl;// 띄어쓰기 단위로 example.txt에서 읽어와서 출력  //}  fin.close();  } |
|  |

3. 아래의 프로그램을 작성하시오.(/\*구현\*/ 부분을 채울 것)

|  |
| --- |
| #include <iostream>  using namespace std;  int main() {  double d1 = 1.23456789;  cout << d1 << endl;  /\*구현\*/ // 출력값 길이 10로 지정  /\*구현\*/ // 주요 자리수 3자리만 출력  cout << d1 << endl;  char ch1;  char ch2;    // ch1, ch2 : enter “ z”. ( space + z )  cin >> ch1;  cout << “(“ << ch1 << “)” << endl;  cin.ignore(); // enter가 다음 cin으로 들어가지 않도록 해줌  /\*구현\*/ // space도 입력으로 생각하게 함  cin >> ch2;  cout << “(“ << ch2 << “)” << endl;  return 0;  } |
|  |
| #include <iostream>  using namespace std;  int main() {  double d1 = 1.23456789;  cout << d1 << endl;  cout.width(10);  cout.precision(3);// 주요 자리수 3자리만 출력  cout << d1 << endl;  char ch1;  char ch2;  // ch1, ch2 : enter “ z”. ( space + z )  cin >> ch1;  cout << "(" << ch1 << ")" << endl;  cin.ignore(); // enter가 다음 cin으로 들어가지 않도록 해줌  cin.unsetf(ios::skipws); // space도 입력으로 생각하게 함  cin >> ch2;  cout << "(" << ch2 << ")" << endl;  return 0;  } |
|  |

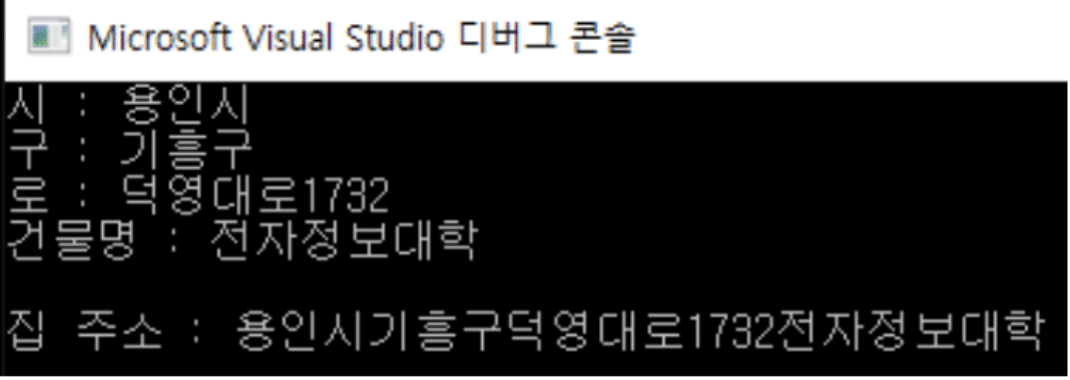
4. 아래의 프로그램을 작성하시오.(/\*구현\*/ 부분을 채울 것)

|  |
| --- |
| #include <iostream>  #include <fstream> //ifstream ofstream  #include <iomanip>  using namespace std;  bool getStu(/\*구현\*/) {  fin >> id >> exam1 >> exam2 >> exam3;  if (!fin)  return false;  return true;  }  void calcAvgGrade(/\*구현\*/) {  avg = (exam1 + exam2 + exam3) / 3;  if (avg >= 90)  grade = 'A';  else if (avg >= 80)  grade = 'B';  else  grade = 'F';  }  void writeStu(/\*구현\*/) {  fout.fill('0');  fout << setw(4) << id;  fout.fill(' ');  fout << setw(4) << avg;  fout << setw(4) << grade << endl;  }  int main() {  //반복: 모든 학생을 읽고 저장할 때까지  //getStu 함수 = ch7STUFL.DAT에서 파일 읽기: id, exam1, exam2, exam3  //calcAvgGrade 함수 = 평균,grade 계산: exam1, exam2, exam3 -> avg, grade  //writeStu 함수 = grade.txt 저장: id, avg, grade  /\*구현\*/ fin("ch7STUFL.DAT"); // 강의 사이트에서 다운로드 가능  /\*구현\*/ fout("grade.txt");  int id, exam1, exam2, exam3, avg;  char grade;  while (/\*구현\*/) { // getStu함수 사용  /\*구현\*/ // calcAvgGrade함수 사용  /\*구현\*/ // writeStu함수 사용  }  fin.close();  fout.close();  cout << "end";  return 0;  } |
| **grade.txt 파일 입력 결과** |
| #include <iostream>  #include <fstream> //ifstream ofstream  #include <iomanip>  using namespace std;  bool getStu(ifstream& fin, int& id, int& exam1, int& exam2, int& exam3 ) {  fin >> id >> exam1 >> exam2 >> exam3;  if (!fin)  return false;  return true;  }  void calcAvgGrade(int exam1, int exam2, int exam3, int& avg, char& grade ) {  avg = (exam1 + exam2 + exam3) / 3;  if (avg >= 90)  grade = 'A';  else if (avg >= 80)  grade = 'B';  else  grade = 'F';  return;  }  void writeStu(ofstream& fout, int id, int avg, char grade) {  fout.fill('0');  fout << setw(4) << id;  fout.fill(' ');  fout << setw(4) << avg;  fout << setw(4) << grade << endl;  return;  }  int main() {  //반복: 모든 학생을 읽고 저장할 때까지  //getStu 함수 = ch7STUFL.DAT에서 파일 읽기: id, exam1, exam2, exam3  //calcAvgGrade 함수 = 평균,grade 계산: exam1, exam2, exam3 -> avg, grade  //writeStu 함수 = grade.txt 저장: id, avg, grade  ifstream fin("ch7STUFL.DAT"); // 강의 사이트에서 다운로드 가능  ofstream fout("grade.txt");  int id, exam1, exam2, exam3, avg;  char grade;  while (getStu(fin, id, exam1, exam2, exam3)) { // getStu함수 사용  calcAvgGrade(exam1, exam2, exam3, avg, grade);// calcAvgGrade함수 사용;  writeStu(fout, id, avg, grade); // writeStu함수 사용  }  fin.close();  fout.close();  cout << "end";  return 0;  } |
|  |

**<응용문제>\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. 집 주소(string address)를 시, 구, 로, 건물명(string city, area, street, building) 각각 입력받은 뒤 출력하는 프로그램을 작성하라. (단, 출력은 하나의 string 변수로 출력한다)

1-출력화면:

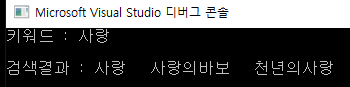


|  |
| --- |
| #include <iostream>  #include <string>  using namespace std;  int main() {  string Si, Gu, Ro, BdName, adress;  cout << "시 : ";  cin >> Si;  cout << "구 : ";  cin >> Gu;  cout << "로 : ";  cin >> Ro;  cout << "건물명 : ";  cin >> BdName;  cout << endl;  adress = Si + Gu + Ro + BdName;  cout << "집 주소 : " << adress;  } |
|  |

2. 검색엔진을 만들려고 한다. 특정 키워드를 입력했을 때 완전히 일치하는 단어뿐만 아니라 키워드를 포함하는 단어까지 전부 출력해라. ( hint-> string의 find() 함수 사용 )

|  |
| --- |
| 데이터 베이스 : string data = "사랑,프로그래밍,의자,사랑의바보,영통역,천년의사랑,냉장고,  객체지향"; 와 같은 형태로 생성  검색엔진의 데이터베이스에 있는 단어 : 사랑, 프로그래밍, 의자, 사랑의바보, 영통역, 천년의사랑, 냉장고, 객체지향  키워드 : 사랑  검색결과 : **사랑**  **사랑**의바보 천년의**사랑** |

2 – 출력화면 :

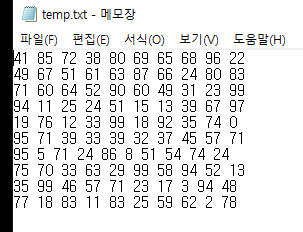


|  |
| --- |
| #include <iostream>  #include <string>  using namespace std;  int main() {  string data = "사랑, 프로그래밍, 의자, 사랑의바보, 영통역, 천년의사랑, 냉장고, 객체지향";  string key, s1;  cout << "키워드: ";  cin >> key;  cout << "검색결과 : ";  int i = 0;  while (1) {  s1.clear();  int index = data.find(key, i);  int ref = index;  if (index == string::npos)  break;  if (index == 0) {  while (1) {  if (data.at(ref) == ',') {  i = ref;  break;  }  else {  s1 = s1 + data.at(ref);  ref++;  }  }  }  else {  while (data.at(ref) != ' ') {  ref--;  }  while (1) {  ref++;  if (data.at(ref) == ',') {  i = ref;  break;  }  else s1 += data.at(ref);  }  }  if (data.find(key, i) != string::npos)  cout << s1 << ", ";  else  cout << s1;  }  } |
|  |

3. 0~100 사이의 정수를 랜덤하게 10x10 행렬로 만들고 txt파일을 생성해 그 결과를 확인할 수 있는 프로그램을 작성하라.

|  |
| --- |
|  |

3 – 출력화면 :

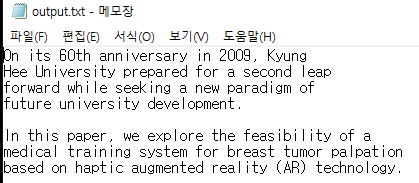


|  |
| --- |
| #include <iostream>  #include <fstream>  using namespace std;  int main()  {  ofstream fout;  fout.open("temp.txt");  int num;  for(int i = 0; i < 100; i++)  {  num = rand() % 101;  fout << num << " ";  if (i % 10 == 9)  fout << endl;  }  fout.close();  return 0;  } |
|  |

4. 아래와 같은 두개의 다른 텍스트파일을 하나의 텍스트 파일로 합치는 프로그램을 작성하라. (단, 줄바꿈도 텍스트 파일 형식에 포함된다)

|  |  |  |  |
| --- | --- | --- | --- |
| 두개의 텍스트 파일   |  |  | | --- | --- | | On its 60th anniversary in 2009, Kyung  Hee University prepared for a second leap  forward while seeking a new paradigm of future university development. | In this paper, we explore the feasibility of a  medical training system for breast tumor palpation  based on haptic augmented reality (AR) technology. |   합친 하나의 텍스트파일   |  | | --- | | On its 60th anniversary in 2009, Kyung  Hee University prepared for a second leap  forward while seeking a new paradigm of  future university development.  In this paper, we explore the feasibility of a  medical training system for breast tumor palpation  based on haptic augmented reality (AR) technology. | |

4 – 출력화면 :



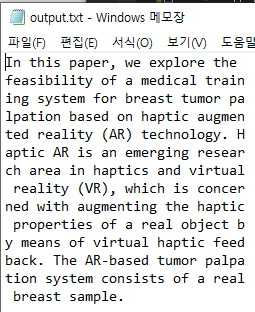
|  |
| --- |
| #include <iostream>  #include <fstream>  #include <string>  using namespace std;  int main()  {  ifstream fin;  ifstream fin2;  ofstream fout;  fin.open("fileA.txt");  fout.open("output.txt");  if (fin.is\_open() && fout.is\_open()) {  string line;  while (getline(fin, line)) {  fout << line << endl;  }  }  fin.close();  fout << endl;  fin.open("fileB.txt");  if (fin.is\_open() && fout.is\_open()) {  string line;  while (getline(fin, line)) {  fout << line << endl;  }  }  fin.close();  fout.close();  return 0;  } |
|  |

5. 아래와 같은 텍스트파일을 읽고 각 라인의 최대 길이를 입력 받은 후 새로운 텍스트 파일을 만드는 프로그램을 작성하라.

|  |  |  |
| --- | --- | --- |
| 기존 텍스트 파일   |  | | --- | | In this paper, we explore the feasibility of a  medical training system for breast tumor palpation based on haptic  augmented reality (AR) technology. Haptic AR is an emerging  research area in haptics and virtual reality  (VR), which is concerned  with augmenting the haptic  properties of a real object by means of virtual haptic  feedback. The AR-based tumor palpation system  consists of a real breast sample. |   새로운 텍스트 파일 (length=35)   |  | | --- | |  | |

5 – 출력화면 :





|  |
| --- |
| #include <iostream>  #include <fstream>  #include <string>  using namespace std;  int main()  {  int len = 200;  cout << "length = ";  cin >> len;  ifstream fin;  ofstream fout;  fin.open("fileA.txt");  fout.open("output.txt");  if (fin.is\_open() && fout.is\_open()) {  string line, s1="";  while (getline(fin, line)) {  s1 += line;  s1 += " ";    }  int i=0;  while (i < s1.length()) {  fout.width(len);  fout << s1.substr(i, len) << endl;  i += len;  }  }  fin.close();  fout.close();  return 0;  } |
|  |