1.

(A) msg = "hello";	[2]
(B) msg = msg + "hello"; (msg +=)	[2]
(C) legth = msg.len(); (.length();)	[2]

2.

(A) const int argc, Const char[] agv (char* argv[])	[2]
(B) int i = 0; i < argc ; i++ (1) 問題見ろね	[2]
(C) String str = argv[i];	[2]
(D) str.reverse(); (reverse(str.begin(), str.end());)	[2]

3.

```
(略)
class Point {
  int x, y, z;
public:
  friend istream& operator>>(istream& in, Point& p);
  friend ostream& operator<<(ostream& out, const Point& p);

};
istream& operator>>(istream& in, Point& p) {
    in >> p.x >> p.y >> p.z;
    return in;
}
ostream& operator<<(ostream& out, const Point& p){
    out << p.x << ' ' << p.y << ' ' << p.z;
    return out;
}

int main() { (略) }
```

[6] [8]

4.

```
#include <iostream>
#include <fstream>                          (A) [2]

using namespace std;

int main()
{
  int n;           // 各行の「個数」
  float w;         // 各行の「重さ」
  float total = 0; // 総重量

  ifstream in;                              (B) [2]
  in.open("data.txt");

                                   (右上へ)
while (!in.eof()){                          (C) [8]
    in >> n >> w;
    total += static_cast<float>(n*w);
}

ofstream out;                               (D) [2]
out.open("output.txt");
out << total;                               (E) [2]
}
```

256

5．

（ア） #include <iostream> ←バカアホ

#include <iomanip>

[2]

（イ）（枠内に一文字）

f	f	f	0	1	2										[3]
f	f	f	0	f	f	f	1	f	f	f	2				[3]
0	x	1	0	0											[3]

6．

（ア） int Fuga::val2; [2]

（イ）
sizeof(Hoge)= 8 ，sizeof(Fuga)= 4 ［完答 2］

（ウ）
hoge.val1， hoge.val2，
fuga1.val1， fuga1.val2，
fuga2.val1， fuga2.val2，
Fuga::val2
［完答 3］

7．

（ア）[BallFactory.cpp]

```
#include "Ball.h"
#include "BallFactory.h"
#include "BasketBall.h"
#include "TennisBall.h"
#include "BaseBall.h"

Ball* BallFactory::Create(Ball::TYPE type) {
    switch (type) {
        case BASKETBALL:
            return new BasketBall;
        case BASEBALL:
            return new BaseBall;
        case TENNISBALL:
            return new TennisBall;
        default:
            return nullptr;
    }
    return nullptr;
}
```

[10]

（イ）[BallApp.cpp]

```
void BallApp::init()
{
  balls.resize(100);

  for (auto& b : balls) {
    int r = Rand::range(0, 9); // 0～9の乱数
    if (r <1)
        b = BallFactory::Create(Ball::TYPE::BASKETBALL);
    else if(r <4)
        b = BallFactory::Create(Ball::TYPE::BASEBALL);
    else
        b = BallFactory::Create(Ball::TYPE::TENNISBALL);

    b->init();
  }
}
```

[8]

8．

アイエ [4]

9．

イオ [4]

10．

（ア） 出力ストリーム [2]	（イ） シングルバイト文字 [2]	（ウ） map 連想配列 [2]
（エ） 区切り文字 [2]	（オ） ならってらん [2]	（カ） 挿入演算子 [2]

抽出

１１．

```
#include <iostream>
#include <stdexcept>                                          [2]

using namespace std;
class difference_size_error : public logic_error {            [3]
public:
  difference_size_error(const string& e): logic_error(e){}
};

class Vector {
  // 略
  Vector operator+(const Vector& v) {
    if ( size != v.size ) {                                   [3]
      throw difference_size_error("２つのベクトルのサイズが違います");
    }

    Vector tmp(size);
    for (size_t i = 0; i < size; i++)
      tmp.vec[i] = vec[i] + v.vec[i];
    }
  // 略
};

int main()
{
  Vector v1(10);
  Vector v2(8);
  Vector v3;
  try {                                                       [3]
  v3 = v1 + v2;
  }
  catch(const exception& e){
    cout << e.what() << endl;
  }
  // 以降何かの処理
}
```

１２．

空欄(A) [3] gu::speak();
空欄(B) [3] using namespace tyoki;
空欄(C) [3] pa::speak();