株式会社ナレッジフロー 社員採用向け 入社試験

問1

問1-1
以下のサイトは、文章を投稿する機能を備え、投稿された文章の最新の20件を表示するWebサイトです。

https://exam.1-mise.net/test1/test1

C#で簡単な画面を作り、その画面で入力した文章を、上記サイトに投稿するプログラムを作成してください。

問1-2
以下のサイトに対し、問1-1と同様のことを行うプログラムを作成してください。

https://exam.1-mise.net/test2/

問1-3
以下のサイトに対し、問1-1と同様のことを行うプログラムを作成してください。

https://exam.1-mise.net/test3/

【注意事項】
通信には System.Net.Http.HttpClient を使用してください。（ブラウザコンポーネントを使用して上記サイトをブラウザコンポーネントで表示すれば同じことができますがその方法はとらないでください。）

【備考】
Webサイトへの投稿は自由に行っていただいて構いません。

問2

C#とAngleSharpを使って以下のプログラムを作成してください。

Wordpressにアクセスし、ログインを行った上で新規記事を作成します。
作成する記事のタイトルおよび本文は、「テスト」などの適当な内容で構いません。

ログイン情報(ログイン画面 URL、ログイン ID、パスワード)は下記のとおりです。

- ログイン画面URL
  http://2ch.secret.jp/test4/wp-admin/
- ログインID
  KF_programmer
- ログインパスワード
  bJacDuBMSe

ログイン後の新規記事の作成方法は以下のとおりです。

1)
ログイン後、画面左のメニューから、「投稿」 > 「新規追加」 と進みます。

2)
記事の新規投稿画面に移ります。
タイトル部分に記事タイトルを、本文入力欄に記事本文を入力します。

3)
画面右の「公開」ボタンを押すことで新規記事を投稿できます。

問3

以下のJavaScriptの２つのプログラムを実行したときに、result1, result2にセットされる値が異なるようなAとBの具体例を作ってください。

```
let result1 = A(function() {
                  return (function(x) {
                    return x;
                  })(B());
                });


let result2 = (function(x) {
                 return A(function() {
                   return x;
                 });
               })(B()));
```

問4

【予備知識】JavaScriptの Promise, then, catch について学習してください。

問4-1

以下のプログラム11とプログラム12の実行が異なるような関数 f11, f12, f13, f14 の具体例を作ってください。

・プログラム11

```
const p11 = new Promise((resolve, reject) =>
      { setTimeout(() => {f11();f12(); resolve();}, 1000) });

p11.then(() => {f13();}).catch(f14);
```

・プログラム12

```
const p12 = new Promise((resolve, reject) =>
      { setTimeout(() => {f11(); resolve();}, 1000) });

p12.then(() => {f12();f13();}).catch(f14);
```

問4-2

以下のプログラム21とプログラム22の実行結果が異なるようなプロミス p21 と、関数 f21, f22 の具体例を作ってください。

・プログラム21

```
p21.then(f21).catch(f22)
```

・プログラム22

```
p21.then((value) => {f21(value);}).catch(f22)
```

問5

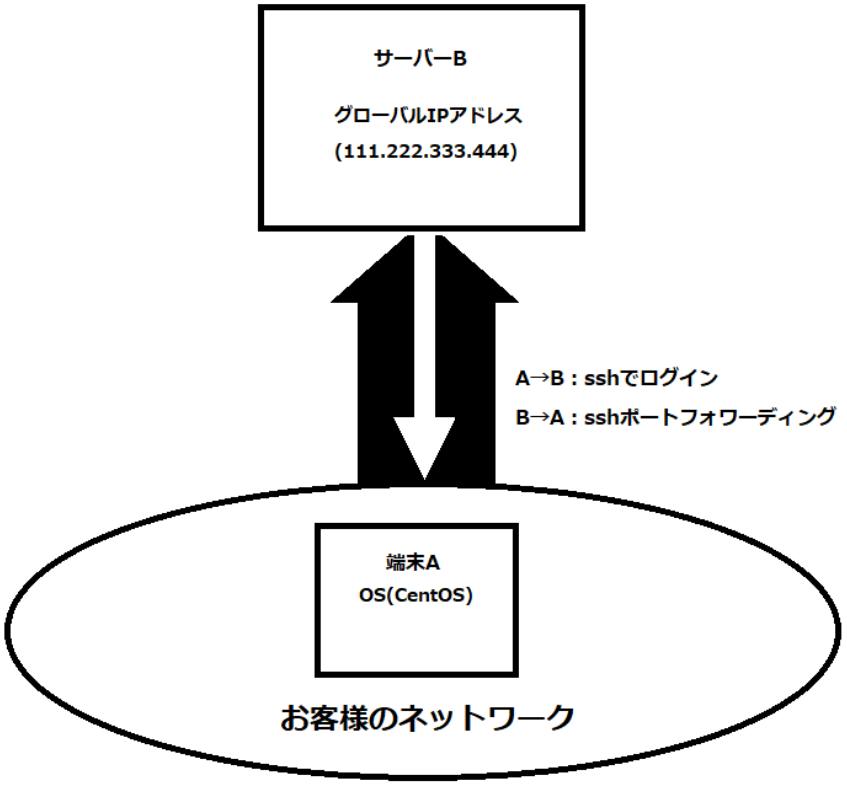
Eメールを受信する機能を備え、本日のEメール受信件数と本文の文字数合計を表示するWebサイトを作成してください。

参考サイト
https://exam.1-mise.net/test1/test4

問6

CentOSがインストールされている端末Aを、お客様のネットワークに設置しているとします。

端末Aをリモートでメンテナンスするために、グローバルIPアドレス（111.222.333.444）を持つサーバーB（OSはCentOS）から端末Aにsshでログインすることを考えます。具体的には、端末Aからサーバー Bにsshログインし、その接続を使ってsshポートフォワーディングでサーバーBから端末Aにsshでログインすることを考えます。



端末Aが再起動したり、端末Aとサーバー Bでの間のネットワークに一時的な断線が起こった場合でも、再起動後もしくはネットワーク復旧後に自動的に上記のリモートメンテナンスの仕組みが復旧するようにするために、どのように設定しますか？（もしくは、どのようなソフトウェアをインストールしますか？／どのようなスクリプトを作成しますか？）具体的に記してください。

問7

Android端末（バージョン9）を支給します。
Android端末をroot化してください。

【注意事項】
Android端末のroot化の情報サイトには、詐欺系の情報が多いので気を付けてください。
PCがウイルスに感染しないように十分に自衛をした上で、取り組んでください。

問題8

以下は、Androidで通話を自動的に録音しファイルに保存するプログラムのソースコードです。

```
public class MainActivity extends AppCompatActivity {
    private static final int PERMISSION_REQUEST_CODE = 12;
    private MediaRecorder recorder = null;
    private String filePath = Environment.getExternalStorageDirectory() + "/sample.mp4";

    private PhoneStateListener phoneStateListener = new PhoneStateListener() {
        @Override
        public void onCallStateChanged(int state, String phoneNumber) {
            switch (state) {
                case TelephonyManager.CALL_STATE_IDLE:
                    try {
                        if (recorder != null) {
                releaseRecorder();
                        }
                    } catch (Exception e) {
                        e.printStackTrace();
                    }
                    break;

                case TelephonyManager.CALL_STATE_OFFHOOK:
                    try {
                       if (recorder != null) {
                            releaseRecorder();
                        }
                        recorder = new MediaRecorder();
                        recorder.setAudioSource(MediaRecorder.AudioSource.DEFAULT);
                        recorder.setOutputFormat(MediaRecorder.OutputFormat.MPEG_4);
                        recorder.setAudioEncoder(MediaRecorder.AudioEncoder.DEFAULT);
                        recorder.setOutputFile(filePath);
                        recorder.prepare();
                        recorder.start();
                    } catch (Exception e) {
                        e.printStackTrace();
                    }
                    break;

                case TelephonyManager.CALL_STATE_RINGING:
                    break;
            }
        }
    };

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        List<String> permissionList = new ArrayList<>(Arrays.asList(
                Manifest.permission.READ_PHONE_STATE,
                Manifest.permission.RECORD_AUDIO,
                Manifest.permission.WRITE_EXTERNAL_STORAGE
```

```
        ));

        String[] permissions = permissionList.toArray(new String[0]);
        if (!hasSelfPermissions(this, permissions)) {
            ActivityCompat.requestPermissions(this, permissions, PERMISSION_REQUEST_CODE);
        }

        TelephonyManager telephonyManager = (TelephonyManager)
getSystemService(Context.TELEPHONY_SERVICE);
        telephonyManager.listen(phoneStateListener, PhoneStateListener.LISTEN_CALL_STATE);
    }

    private boolean hasSelfPermissions(Context context, String... permissions) {
        if (Build.VERSION.SDK_INT < Build.VERSION_CODES.M) {
            return true;
        }
        for (String permission : permissions) {
            if (context.checkSelfPermission(permission) != PackageManager.PERMISSION_GRANTED)
{
                return false;
            }
        }
        return true;
    }

   private void releaseRecorder() {
        recorder.stop();
        recorder.reset();
        recorder.release();
        recorder = null;
    }
}
```

このプログラムはAndroidバージョン8では正しく動作していました。

このプログラムを最近のAndroidで動かす場合にどういう問題や制約・制限などがあるか調べ、調べた結果をまとめてください。