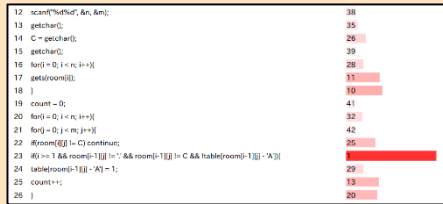


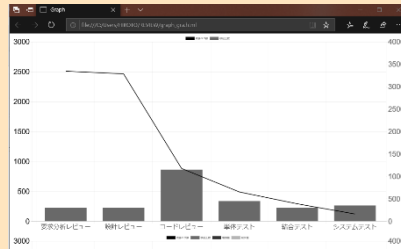
ソフトウェア分析学研究室の 研究テーマ

4つの研究分野

開発プロジェクト支援



バグ予測

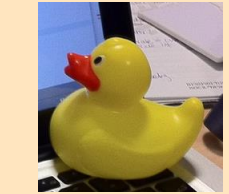


ソフトウェア開発 シミュレータ

開発者支援

ラバーダックデバッグの 実証実験

プログラマ



ラバーダック

ヒューマンコンピュータ インタラクション



VRによる遠隔授業システム

ソフトウェアセキュリティ

```
int func(int c)
{
    return c - 'A' + 'a';
}
```

元のプログラム

```
int func(int c){int c1, c2,
x1=6, x2=13; c1=(c+x1)%
13; c2=(c+x2)%19; return
decode(c1, c2);}
int decode(int a, int b){ int
x=(a+13*(b-a)*3)%247;
if(x<0) x+= 247; return x;}
```

難読化後のプログラム

プログラムの難読化

バグ位置の推定

■ 機械学習により, プログラム中のバグ位置を推定し, デバッグに役立ってます.

```
int check_bendchange(int p)
{
    static int last_bend = -1;
    if(p == -1){
        last_bend = -1;
        return -1;
    }
    if(p == last_bend) return -1;
    last_bend = p;
    return 0;
}
```

ソースコード

機械学習



バグ予測モデル



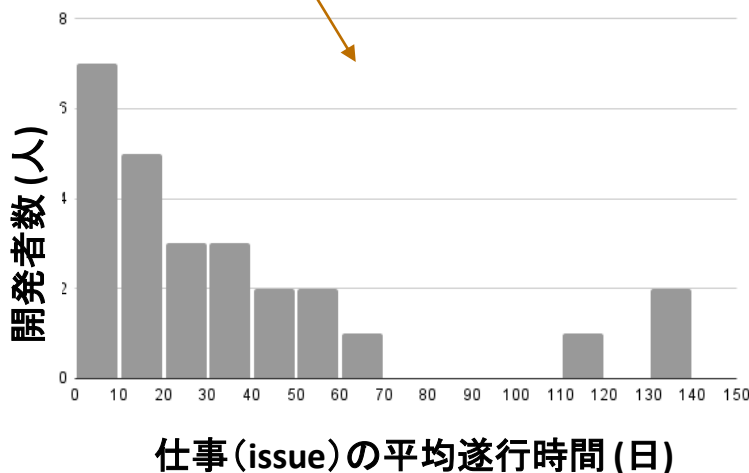
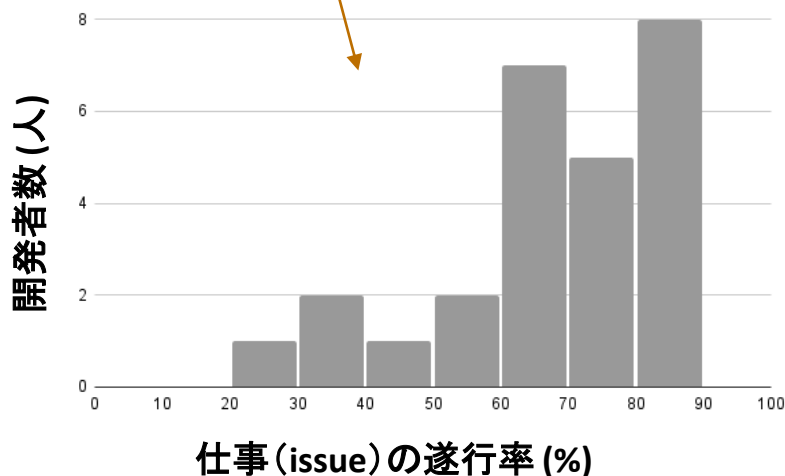
各行のバグ含有確率

15	getchar();	39
16	for(i = 0; i < n; i++){	28
17	gets(room[i]);	11
18	}	10
19	count = 0;	41
20	for(i = 0; i < n; i++){	32
21	for(j = 0; j < m; j++){	42
22	if(room[i][j] != C) continue;	25
23	if(i >= 1 && room[i-1][j] != '.' && room[i-1][j] != C && !table[room[i-1][j] - 'A']){	1
24	table[room[i-1][j] - 'A'] = 1;	29
25	count++;	13
26	}	20

ソフトウェア開発者の信頼性の分析

■ GitHub上の開発者を分析し、**信頼される開発者に求められる行動特性**を分析した。

- 仕事が丁寧である.
- 仕事のスピードが速い.
- 仕事を投げ出さない.
- 主体的に行動する.



React Project

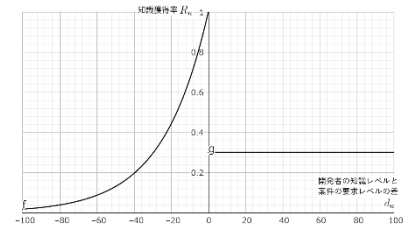
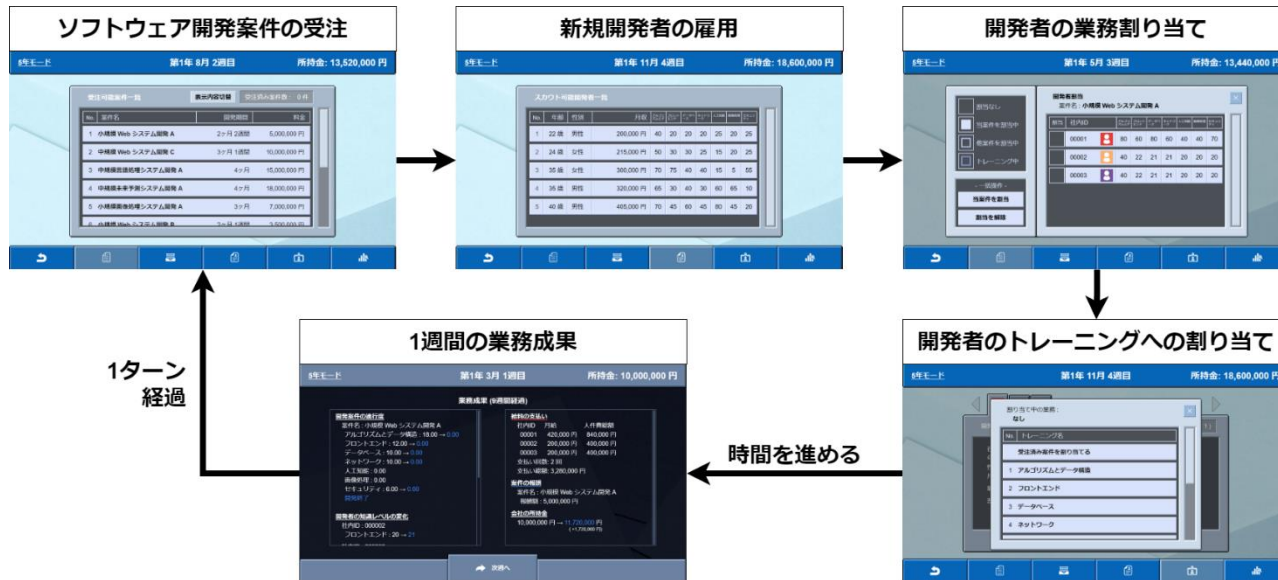
信頼に値しない開発者は少ない。

(ただし、そのような開発者にも仕事が多く割り当てられている実態が明らかとなった。)

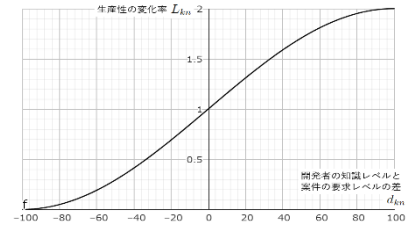
開発管理シミュレーションゲームの開発

<https://www.okayama-u.ac.jp/user/salab/spm/>

■ 経営者・プロジェクト管理者として会社を成長させる。



知識モデル

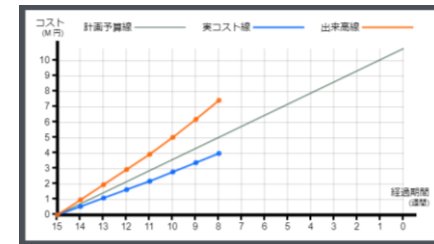


生産性モデル

■ 「人月」に加えて、各開発者の**技術レベル**と**習熟**を考慮したマネジメントを学ぶ。

■ **Earned Value Management (EVM)**によるコスト管理を学ぶ。

■ 開発見積もり、生産性に影響する要因を理解する。



EVM

ラバーダックデバッグの効果の実証

■ プログラムを他者(ラバーダック)に説明しているうちにバグに気が付く。

ねえ、アヒルさん。
ここでxにRが代入されて
関数Yが呼ばれるんだよ。

そんなことアヒル
には分らんがな。



ラバーダック



プログラマ

被験者	個人レビュー	説明作業	他者(説明対象)
A	5	2	人間
B	2	2	人間
C	8	0	人間
D	9	4	人間
E	2	3	人間
F	8	2	人間
G	7	3	ラバーダック
H	2	1	ラバーダック
I	10	2	ラバーダック
J	12	3	ラバーダック

10名中9名で効果あり

バーチャルリアリティ(VR)による 授業視聴システムの開発



授業を3Dカメラで撮影



それをVRゴーグルを用いて見る

使用性を向上させるための改良

- 3つのカメラによる録画と合成
 1. 教室用カメラ: 臨場感を与える.
 2. 黒板用カメラ: 拡大・縮小できる.
 3. 卓上カメラ: 手元のノートに記入できる.

岡山理科大学附属中学校の授業に導入し、評価を行ったところ、好評であった。

スマホゲームにおける課金誘導方法の分析

- スマホゲームに対する過度の課金が社会問題となっている。
- 代表的なスマホゲームを分析し、課金誘導方法の実態を明らかにする。

主な課金誘導方法

- **ガチャ**：報酬のランダム性がギャンブル依存をもたらす。
- **お得な課金**：期間限定，回数限定，初心者限定，〇〇パック
- **キャラクタ**の所有，強化
- **プレイ時間短縮**
- **上限撤廃**
- **アツい演出**：感情の揺さぶり

	モンスト	ドカバト	パズドラ	パワプロ	ポケGO	まおりゅう	でんでん	バンドリ	ウマ娘
ガチャ	○	○	○	○		○	○	○	○
お得な課金	○	○	○	○	○	○	○	○	○
上限撤廃	○	○	○	○	○	○		○	○
時間短縮	○	○	○	○	○	○	○	○	○
キャラ成長	○	○	○	○	○	○	○	○	○
アツい演出	○	○	○	○		○	○	○	○

多くのゲームは、課金誘導方法の大部分を含んでいる実態が明らかとなった。

発注仕様書におけるセキュリティ要件の分析

- 近年，情報システムに対するサイバー攻撃の脅威が増大しているが，今日の情報システムにどの程度のセキュリティ要件が盛り込まれているかは不明である。
- 地方自治体，病院，図書館等の情報システムの発注仕様書（提案依頼書）を分析した。

中項目	自治体	病院	図書館	大学
前提条件・制約条件	0.7	0.9	0.4	0.6
セキュリティリスク分析	0	0	0	0.2
セキュリティ診断	0	0	0	0.3
セキュリティリスク管理	0.7	0.4	0.6	0.8
アクセス・利用制限	1	1	0.9	1
セキュリティインシデント対応・復旧	0.3	0.3	0.2	0.5
データの秘匿化	0.6	0.6	0.7	0.8
不正追跡・監視	1	1	0.7	1
ネットワーク対策	0.8	0.5	0.8	0.9
マルウェア対策	0.9	1	0.9	0.9
WEB対策	0	0	0.1	0.4
平均	0.55	0.52	0.48	0.67

「難読化」によるソフトウェア保護

- コンピュータ自動でプログラムを解析する**プログラム自動解析攻撃**が脅威となっています。

- 制約を満たす入力値を得ることができる
- パスワードクラックなどに適用可能。

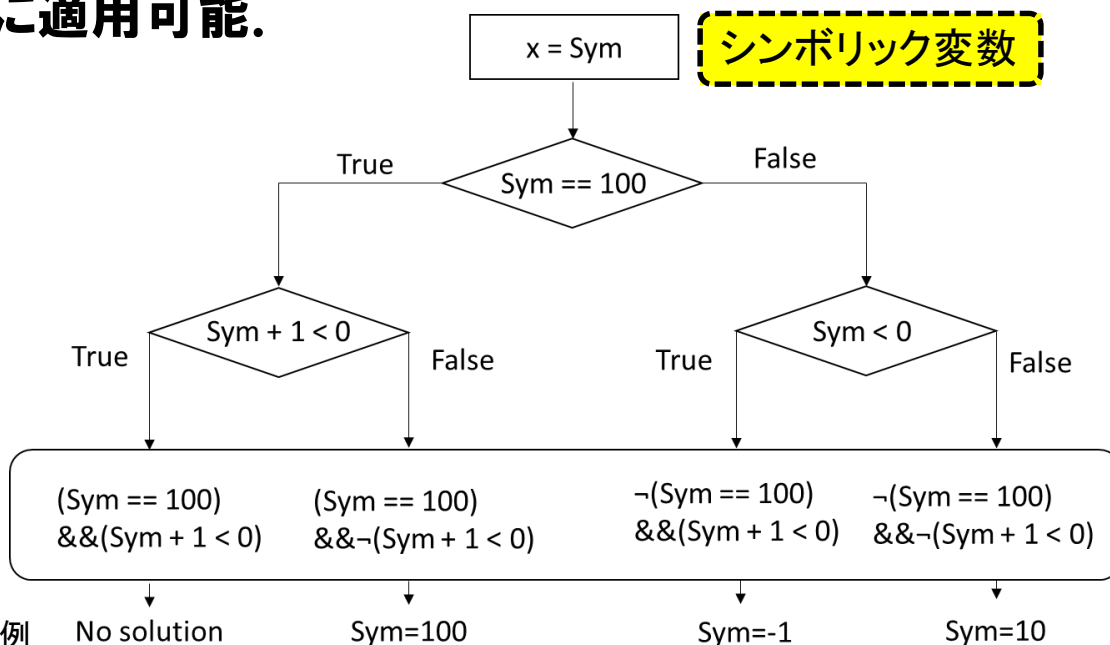
```
int func(int x){  
    if(x == 100)  
        x++;  
    if(x < 0)  
        x = x + 10;  
    return x;  
}
```

プログラム



制約式

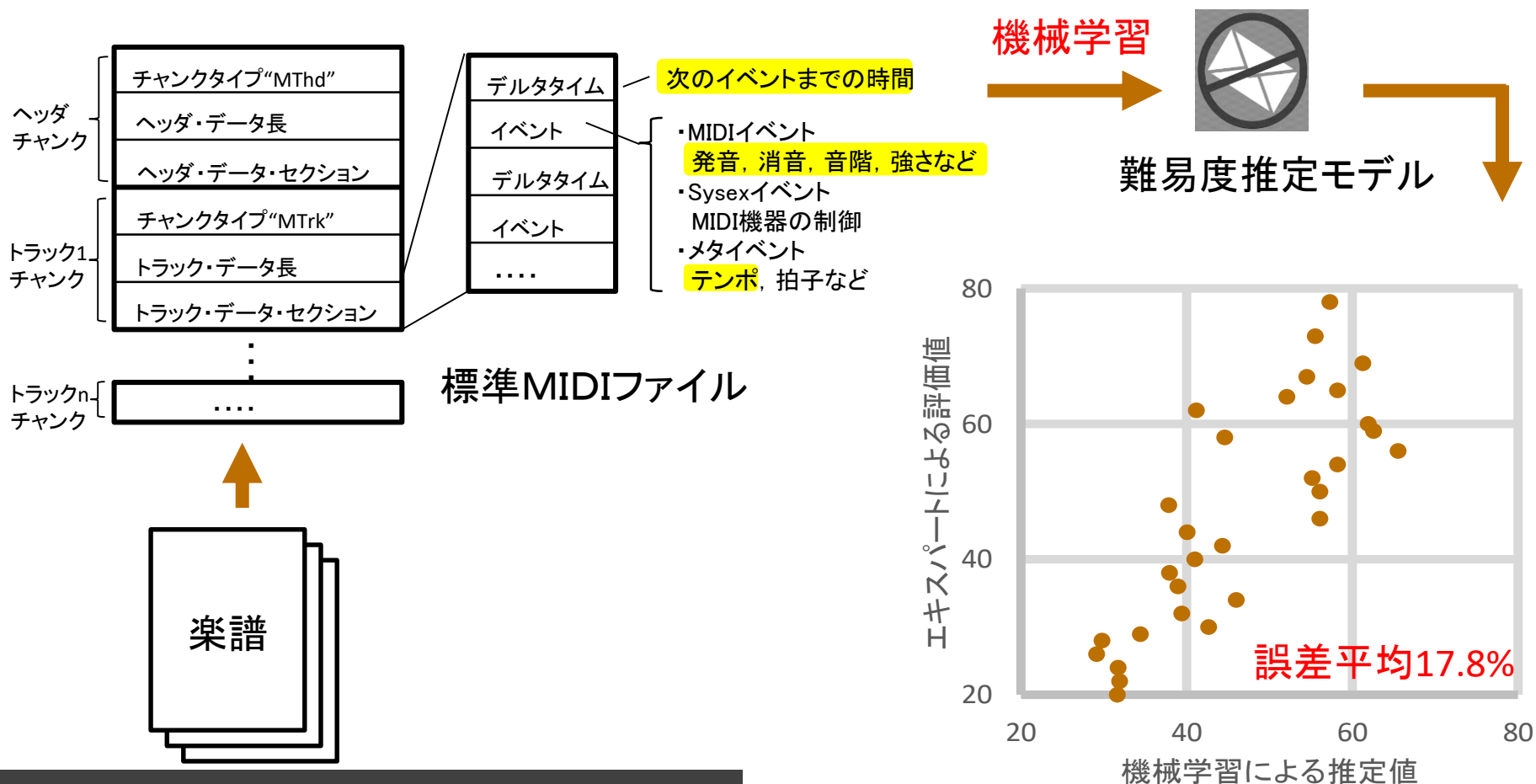
制約ソルバの出力例



- プログラムをコンピュータが解析しにくいものへと変換(難読化)することで、シンボリック実行攻撃を防ぎます。

その他:ピアノ曲の難易度推定

■ 弾きやすさという観点からピアノ曲の難易度の推定を行う。
(芸術性は無視)



2022年度の卒業研究

- スマホゲームにおける課金誘導方法の分析
- 習熟を考慮したソフトウェア開発管理シミュレーションゲーム
- ソフトウェア開発者の信頼度の評価
- ソフトウェアテスト教育のためのテストケース作成実験
- オンライン考古学データベースへの発掘資料のデータ入力補助システムの開発
- 提案依頼書におけるセキュリティ要件の実態調査
- グローバーのアルゴリズムを題材とした学習支援システムの開発

2021年度の卒業研究

- キーストロークとマウス操作に基づくプログラミング能力の分析
- ITエンジニアに求められる技術の求人票に基づく分析
- バグ予測におけるテスト工数削減率に影響する要因の分析
- 料理動画における画像認識を用いたレシピ手順の推定
- 間違い探しにおける視線の分析