

Контролируемое исполнение программ в многозадачном окружении

Защищается: студент группы Б8403А
Ротанов Денис Владимирович
Руководитель: старший преподаватель
Кленин Александр Сергеевич

Спортивное программирование

1. Олимпиады школьников по программированию
2. ACM ICPC (международная студенческая олимпиада)
3. Соревнования искусственного интеллекта

Соревнования ИИ

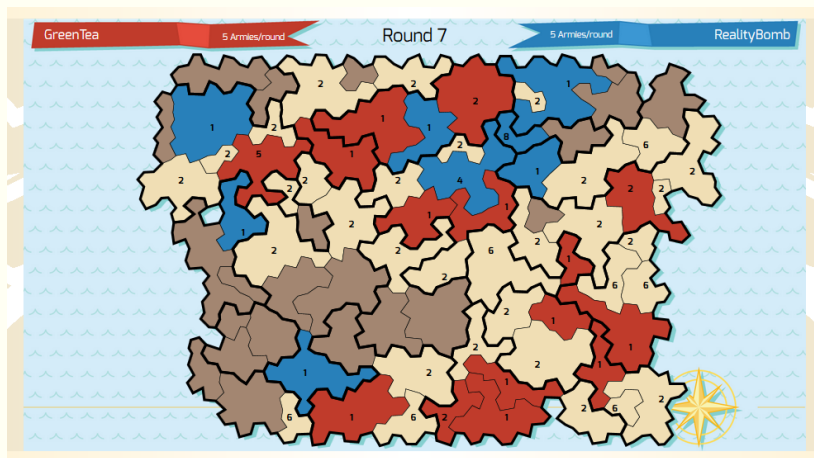
- ▶ виртуальная среда
- ▶ интеллектуальные агенты
- ▶ многозадачность (в смысле процессов ОС)
- ▶ управляющая программа
- ▶ нормальная программа

Соревнования ИИ - Примеры [1]



Vindinium

Соревнования ИИ - Примеры [2]



theaigames - Warlight 2 Challenge

Соревнования ИИ - Примеры [3]



Robocode

Соревнования ИИ - Примеры [4]

The screenshot displays the CodingGame interface, which is divided into four main sections:

- Game View:** Shows a top-down view of a planet being attacked by alien ships. The interface includes a score counter (0), a list of enemies in the zone, a threat level indicator, and a 'VIEW MORE +' button. A status message at the bottom reads: "CodinGame planet is being attacked by slimy insectoid aliens."
- Code editor (C++):** Contains C++ code for a game loop. The code includes headers for `<iostream>`, `<string>`, `<vector>`, and `<algorithm>`. It uses `using namespace std;` and defines a `main()` function with a `while (1) {` loop. Inside the loop, it reads enemy names and distances from standard input and uses `cin.ignore()` to skip the rest of the line. The code is as follows:

```
1 #include <iostream>
2 #include <string>
3 #include <vector>
4 #include <algorithm>
5
6 using namespace std;
7
8 /**
9  * CodinGame planet is being attacked by slimy insectoid aliens.
10  * <---
11  * Hint: To protect the planet, you can implement the pseudo-code provided in
12  */
13 int main()
14 {
15     // game loop
16     while (1) {
17         string enemy1; // name of enemy 1
18         cin >> enemy1; cin.ignore();
19         int dist1; // distance to enemy 1
20         cin >> dist1; cin.ignore();
21         string enemy2; // name of enemy 2
22     }
```
- Console output:** Shows the standard output stream. It displays "Prickster 6m" and "Standard Output Stream: > enemy". Below this, it shows "Game information:" and a red message: "Failure: You were destroyed. You were hit by Bumbleboom".
- Test cases:** A table showing test cases. The first case is labeled "03" and "Surrounded", with a play button icon. A "PLAY ALL TESTS" button is at the top right.

Codingame

CATS - автоматизация соревнований

CATS Archive с начала тура прошло: 5610 суг.
страница обновлена: 12.05.2015 09:28
[выход] [турниры] [задачи] [консоль] [модули] [компиляторы] [ключевые слова] [профиль] [о системе] [регистрация]
[все] [официальные] [незавершенные] [календарь]

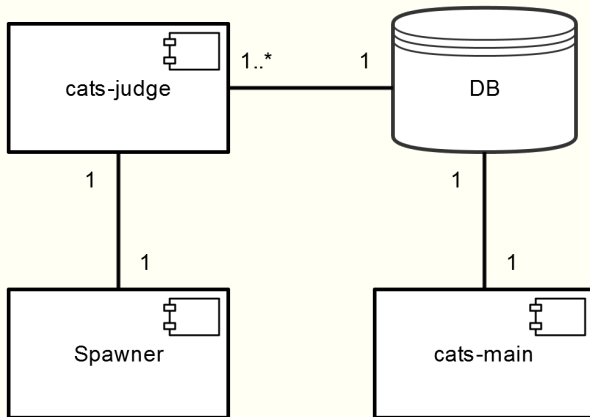
найти: страницы: 1 2 3 4 5 >> 10

Название	Начало	Завершение	Примечание
Archive	01.01.2000 00:00	01.01.2025 00:00	участник
in progress — school	01.01.2001 00:00	31.12.3112 00:00	официальный регистрация закрыта школьные правила
Training, spring 2015	07.05.2015 20:37	07.07.2015 20:37	
68103a, ЯнМТ - ДЗ № 5, весна 2015	27.04.2015 12:20	27.07.2015 12:20	
Students_AdvAlg_Spring_2015	25.04.2015 18:45	25.05.2015 18:45	
Турнир юных программистов 2015	25.04.2015 12:00	25.04.2015 17:01	официальный школьные правила
Динамические ЯП 2015, Руфол 1	18.04.2015 08:35	28.04.2015 08:35	
68103a, ЯнМТ - ДЗ № 4, весна 2015	16.04.2015 10:15	16.07.2015 10:15	
68103a, Программирование на C - ДЗ №1	09.04.2015 10:52	09.07.2015 10:52	
68103a, Основы алгоритмов - ДЗ №3, весна 2015	22.03.2015 21:12	22.07.2015 21:12	

текущий пользователь: Alice предложения | документация | © 2002-2014 Авторы
страница 6/10 Кафедра информатики, математического и компьютерного моделирования
Дальневосточный федеральный университет

CATS - система автоматической проверки решений задач по программированию

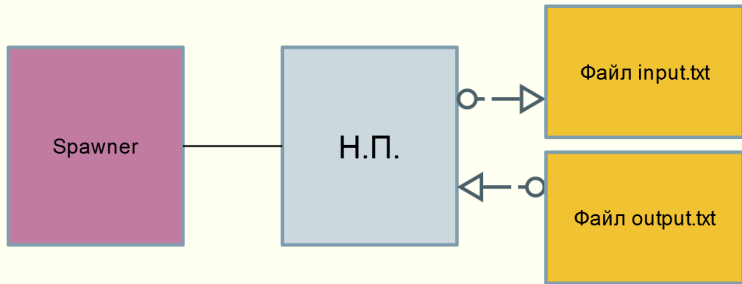
Компоненты CATS



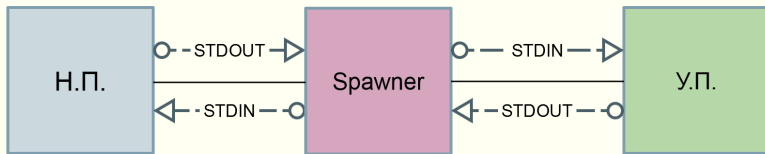
Типы задач

- ▶ Стандартные
- ▶ Интерактивные
- ▶ Многоагентные

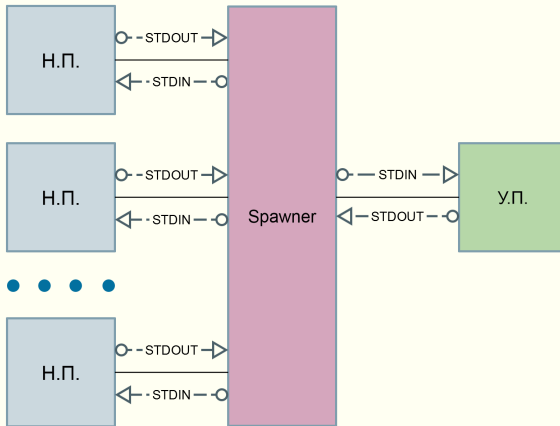
Стандартные задачи



Интерактивные задачи



Многоагентные задачи



Цель работы

Расширение класса задач, поддерживаемых CATS:

- ▶ интерактивные задачи
- ▶ многоагентные задачи

Существующие решения

- ▶ аналоги CATS
 - ✓ интерактивные задачи
 - × многоагентные задачи
- ▶ системы соревнований ИИ
 - ▶ узкоспециализированы (1 задача)
- ▶ средства контролируемого исполнения
 - × связь стандартных потоков
 - × делегирование управления

Потоки ввода-вывода

```
1 #include <stdio>
2 int main()
3 {
4     // STDOUT:
5     printf("Hello World\n");
6
7     // STDIN:
8     int n = -1;
9     scanf("%d", &n);
10 }
```


Реализация интерактивных задач

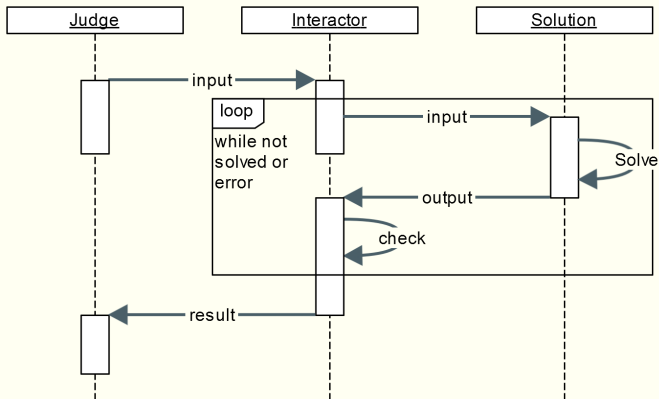
Настройки задачи:

```
1 <Problem title="online addition" lang="ru" tlimit="1"  
2   mlimit="256" inputFile="*STDIN" outputFile="*STDOUT">  
3  
4   <Run method="interactive"/>  
5   ...  
6 </Problem>
```

Настройки cats-judge:

```
1 <define name="#run_interactive" value="#spawn  
2   --separator="// -hr=1 --out=nul -wl=30 -tl=%time_limit  
3   -ml=%memory_limit -y=1 --// -sr=report.txt  
4   --in=*1.stdout --out=*1.stdin %interactor_name  
5   --// %deadline -sr= "/>
```

Интерактивные задачи



Реализация многоагентных задач

- ▶ каждый процесс - два потока
- ▶ n процессов
- ▶ связь одного процесса со многими
- ▶ мультиплексирование потоков
- ▶ протокол

Протокол

- ▶ гарантия атомарности сообщений
- ▶ У.П.: отослать сообщение Н.П.[i]

```
1 printf("i#<message>\n");  
2 fflush(stdout);
```

- ▶ У.П.: завершить Н.П.[i]

```
1 printf("iS#\n");  
2 fflush(stdout);
```

- ▶ У.П.: ожидать сообщения от Н.П.[i]

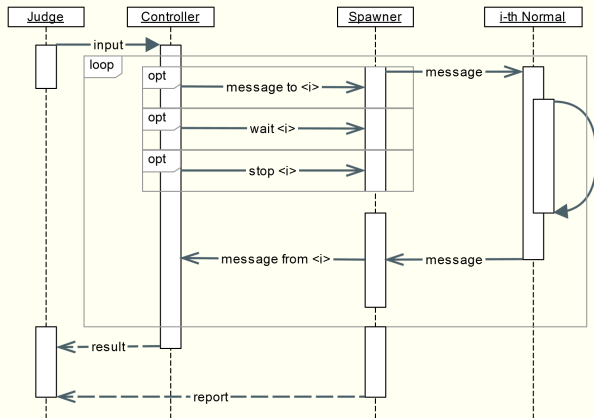
```
1 printf("iW#\n");  
2 fflush(stdout);
```

- ▶ Spawner должен пометить сообщения от Н.П.

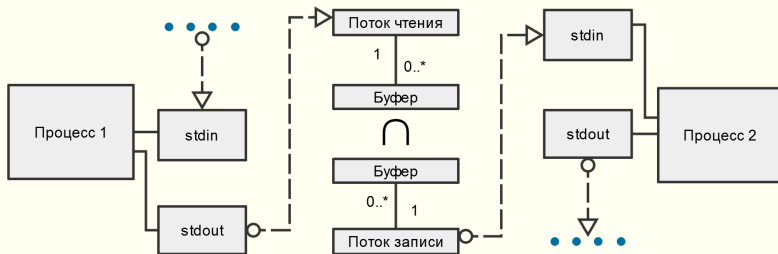
Командная строка запуска Spawner

```
1 sp.exe -tl=1 -d=2 --separator=//  
2 --// -tl=5 -d=5 --controller controller.exe 3  
3 --// --in=*0.stdout --out=*0.stdin normal-1.exe  
4 --// --in=*0.stdout --out=*0.stdin normal-2.exe  
5 --// --in=*0.stdout --out=*0.stdin normal-3.exe
```

Многоагентные задачи



Связь ввода-вывода процессов



Тестирование

- ▶ примеры интерактивных задач
- ▶ примеры многоагентных задач
- ▶ многократный запуск многоагентных задач
- ▶ сборка проекта различными компиляторами C++

Заключение

- ▶ реализована поддержка интерактивных задач в CATS
- ▶ разработан протокол обмена данными для задач
- ▶ модуль Spawner:
 - ▶ реализована поддержка протокола
 - ▶ произведён рефакторинг
 - ▶ исправлены ошибки
- ▶ 4500 строк кода
- ▶ 61 коммит в репозитории проектов CATS