SOPAS - Submissão Online Para Análise de Software

José Pedro Silva Pedro Faria Ulisses Costa

Engenharia de Linguagens
Projecto integrado

March 12, 2011

Index

Até agora:

Concretizado até ao ínicio da segunda fase:

- Descrição do sistema ✓
- Modelação formal e informal do problema ✓
- Modelo de dados ✓
- Ínicio da implementação e respectivo tool demo ✓

Index

Motivação e Objectivos

Objectivos para segunda fase:

- Terminar a aplicação web (compilar e executar o código fonte submetido)
- Investigação das métricas existentes
- Scripts auxiliares
- Exploração de um frontend
- Apresentar resultados

Index

Index

Implementação: até à segunda fase

Já implementado para o último checkpoint:

- Criação de contas de utilizador (grupo)
- Associação de concorrentes a determinado grupo
- Criação de concursos
- Criação de enunciados (através da interface web ou submetendo em formato xml)
- Inserção de baterias de teste para os enunciados
- Submissão de programas para avaliação

Index

Implementação: linguagens de programação

Configuração de linguagens de programação:

- Estando a linguagem correctamente configurada no servidor, é simples preparar o sistema de submissão para avaliar código submetido nessa linguagem
- Para isso basta inserir o comando usado para compilar e para executar, que por exemplo, em C seria:
  String compilação: *gcc -O2 -Wall #{file}*
  String de execução default: *./a.out*
  String de execução para makefile: *./#{file}*

Index

Implementação: Compilação

- Caso seja necessário compilar o código fonte submetido, é usada a string de compilação definida aquando da configuração da linguagem
- Se for submetido um ficheiro comprimido que inclua um makefile, é executado o comando *make* e, o nome do executável criado é obtido a partir de um script perl

Index

Implementação: Execução

- Para executar o programa para os diferentes inputs, é usada a string de execução simples (no caso de ser submetido apenas um ficheiro) ou a string de execução para makefile (no caso de ser submetido um makefile)
- Para cada input o comando é corrido uma vez
- O output é capturado e comparado com o esperado
- É guardada a percentagem de testes no qual o código submetido passou

Implementação: Apresentação de resultados

- A qualquer altura o utilizador pode consultar os resultados das últimas submissões (suas ou dos restantes participantes)
- Pode também consultar os seus melhores resultados, para cada enunciado

Index

Scripts auxiliares

- Script (em Perl) para obter o nome do executável gerado pelo makefile (para C)
- Script (em Perl) que gera estatísticas relativamente à quantidade de ficheiros submetidos para cada linguagem de programação

Script makefile.pl

- Utiliza o módulo perl *Makefile::Parser* para fazer parse do makefile, e obter o nome do executável gerado
- No caso de não ser definido um nome para o output, retorna *a.out*
- TODO: Suportar mais linguagens par além do C.

Script count.pl

- Dada uma pasta, explora recursivamente os seus directórios, e extraí várias estatisticas relativas á quantidade de número de linhas
  - Número de linhas de código por linguagem
  - Número de linhas comentadas por linguagem
  - Rácio entre linhas de código e número de ficheiros para cada linguagem
  - Percentagem de linguagem mais usadas no projecto
  - ...
- Utiliza o módulo perl *GD* para gerar gráficos

Index

Language.C

Index

Terminal Interface

Index

Conclusão e trabalho futuro

- Implementar várias métricas para C com o FrontEnd
- Tornar a interface utilizador mais intuitiva e mais agradável de utilizar

Perguntas

?