



**Program Listing for File
greedy_fragment_allocation.hpp**

[Return to documentation for file \(](#)

`include/holoscan/core/schedulers/greedy_fragment_allocation.hpp`)

```
/* * SPDX-FileCopyrightText: Copyright (c) 2023 NVIDIA CORPORATION & AFFILIATES.
All rights reserved. * SPDX-License-Identifier: Apache-2.0 * * Licensed under the
Apache License, Version 2.0 (the "License"); * you may not use this file except in
compliance with the License. * You may obtain a copy of the License at * *
http://www.apache.org/licenses/LICENSE-2.0 * * Unless required by applicable law
or agreed to in writing, software * distributed under the License is distributed on an
"AS IS" BASIS, * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express or implied. * See the License for the specific language governing
permissions and * limitations under the License. */ #ifndef
HOLOSCAN_CORE_SCHEDULERS_GREEDY_FRAGMENT_ALLOCATION_HPP #define
HOLOSCAN_CORE_SCHEDULERS_GREEDY_FRAGMENT_ALLOCATION_HPP #include
<queue> #include <string> #include <unordered_map> #include <vector> #include
"../fragment_scheduler.hpp" namespace holoscan { class
GreedyFragmentAllocationStrategy : public FragmentAllocationStrategy { public:
void on_add_available_resource(const AvailableSystemResource&
available_resource) override; void on_add_resource_requirement(const
SystemResourceRequirement& resource_requirement) override;
holoscan::expected<std::unordered_map<std::string, std::string>, std::string>
schedule() override; private: struct AvailableSystemResourceComparator { bool
operator()(const AvailableSystemResource& a, const AvailableSystemResource& b)
const; }; struct SystemResourceRequirementComparator { bool operator()(const
SystemResourceRequirement& a, const SystemResourceRequirement& b) const; };
std::priority_queue<AvailableSystemResource,
std::vector<AvailableSystemResource>, AvailableSystemResourceComparator>
available_resources_pq_; std::priority_queue<SystemResourceRequirement,
std::vector<SystemResourceRequirement>,
SystemResourceRequirementComparator> resource_requirements_pq_; }; } //
namespace holoscan #endif/*
HOLOSCAN_CORE_SCHEDULERS_GREEDY_FRAGMENT_ALLOCATION_HPP */
```

© Copyright 2022-2024, NVIDIA.. PDF Generated on 06/06/2024